

ORIENTING DEFORMABLE POLYGONAL PARTS WITHOUT SENSORS

A Thesis

by

SHAWN KRISTEK

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

December 2011

Major Subject: Computer Science

ORIENTING DEFORMABLE POLYGONAL PARTS WITHOUT SENSORS

A Thesis

by

SHAWN KRISTEK

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	Dylan Shell
Committee Members,	Dezhen Song
	Sheng-Jen Hsieh
Head of Department,	Duncan M. Walker

December 2011

Major Subject: Computer Science

ABSTRACT

Orienting Deformable Polygonal Parts without Sensors. (December 2011)

Shawn Kristek, B.S.E.E., University of Texas at Tyler

Chair of Advisory Committee: Dr. Dylan Shell

Parts orienting is an important part of automated manufacturing. Sensorless manipulation has proven to be a useful paradigm in addressing parts orienting, and the manipulation of deformable objects is a growing area of interest. Until now, these areas have remained separate because existing orienting approaches utilize forces that if applied to deformable parts violate the assumptions used by existing algorithms, and could potentially break the part. We introduce a new algorithm and manipulator actions that, when provided with the *geometric description* and a *deformation model* of choice for the part, exploits the deformation and generates a *Plan* that consists of the shortest sequence of manipulator actions guaranteed to orient the part up to symmetry from any unknown initial orientation and pose. Additionally, the algorithm estimates whether a given manipulator is sufficiently precise to perform the actions which guarantee the final orientation. This is dictated by the particular part geometry, *deformation model*, and the *manipulator action path planner* which contains simple *end-effector constraints* and any standard motion planner. We illustrate the success of the algorithm with multiple parts through 192 trials of experiments that were performed with low-precision robot manipulators and six parts made of four types of materials. The experimental trials resulted in 154 successes, which show the feasibility of deformable parts orienting. The analysis of the failures showed that for success the assumptions of zero friction are essential for this work, increased manipulator precision would be beneficial but not necessary, and a simple *deformation model* can

be sufficient. Finally, we note that the algorithm has applications to truly sensorless manipulation of non-deformable parts.

TABLE OF CONTENTS

CHAPTER		Page
I	INTRODUCTION	1
II	PREVIOUS WORKS	5
	A. Sensorless Orienting	5
	B. Manipulation of Deformable Objects	6
III	METHODS	8
	A. The Parts	8
	B. Goldberg's Functions	9
	C. Assumptions	12
	D. Deformable Orientations	13
	E. Adding Deformation	15
	1. Deformation Model	15
	2. End-effector Constraints	16
	a. Push Constraints	17
	b. Grasp Constraints	17
	c. Reorient Constraints	19
	3. Manipulator Action Path Planner	20
	F. Manipulator Actions	22
	G. Algorithm	29
IV	VALIDATION	33
	A. Examples	33
	1. Parts with Executable Plans	33
	2. Parts Requiring Greater Precision	33
	B. Experiments	38
V	CONCLUSIONS	46
	A. Applications to Truly Sensorless Orienting of Rigid Parts	47
	B. Future Work	48
	REFERENCES	50
	VITA	53

LIST OF FIGURES

FIGURE	Page
1	A part represented by the list of vertices of its convex-hull and orientation vector. 8
2	$Diameter(o_i)$ is a plot of the diameter for all orientations o_i . $Squeeze(o_i)$ is a plot of the final orientation after a sweep (squeeze) occurs, given an initial o_i 10
3	$Radius(o_i)$ is a plot of the radius for all orientations o_i . $Push(o_i)$ is a plot of the final orientation after a push occurs, given an initial o_i . 11
4	A <i>deformable orientation</i> of a four-gon part. 13
5	The k levels of <i>deformable orientations</i> ordered from smallest to largest diameter. 14
6	A linear <i>deformation model</i> for an arbitrary part; referred to as a tent model. δ is a deformation distance. β is the angle used to describe excessive deformation. Vol_i is a volume created by incorporating tolerances of the part deformation and is guaranteed to contain the deformed part. Com_i is a volume guaranteed to contain the part's <i>COM</i> 16
7	The PUSH <i>end-effector constraints</i> 18
8	The GRASP <i>end-effector constraints</i> 18
9	The REORIENT <i>end-effector constraints</i> 19
10	Example of reorientation tolerance ϕ for a four-gon part. 20
11	Determining if the manipulator precision is sufficient for reorientation of part o_0 . Fig. 10 showed $\phi = \frac{\pi}{10}$. Manipulator is capable of $\frac{\pi}{2} \pm \frac{\pi}{30}$ REORIENT. 21
12	SWEEP: a primary robot action for reducing possible part poses and orientations. 22

FIGURE	Page
13	Example mappings of SWEEP on possible part orientations of a rectangular part. Shows how parts are reoriented to the <i>deformable orientations</i> , which are at the minima in the <i>Diameter</i> function. 23
14	DEFORM: a primary robot action, a more precise version of SWEEP. 24
15	Example mappings of DEFORM on possible part orientations of a rectangular part. Shows how a DEFORM is a small SWEEP that reorients parts not in <i>deformable orientations</i> and deforms parts by δ_i in <i>deformable orientation</i> o_i 25
16	PUSH is a secondary robot action that reduces possible deformed part poses for GRASP. 26
17	PUSH maps the deformed part to the corner position for grasping. A deformed part in o_i will be at some $\langle X, y_i \rangle$ where $X = [0, 29]$. PUSH will reduce this to $\langle x_0, y_i \rangle$ where x_0 is the corner position. Mappings shown are for four-gon part. 26
18	GRASP: a secondary robot action that precedes REORIENT. 27
19	Reorient: a secondary robot action for mapping deforming orientations to within ϕ of the <i>terminating orientation</i> o_0 27
20	Example mappings of a GRASP followed by REORIENT on the possible orientations of a rectangular part. Shows how a deformed part is mapped to $o_0 \pm \phi$ 28
21	A square part which due to symmetry has $k = 1$ 34
22	A triangular part which has $k = 2$ 35
23	A pentagonal (house) part which has $k = 3$ 36
24	A triangular part for which the manipulators have insufficient precision due to overlapping deltas. 37
25	The complete experimental setup (a), the primary manipulator with one degree-of-freedom in operation space (b), and the secondary manipulator able to PUSH, GRASP, and REORIENT (c). 39

FIGURE	Page
26	The parts used in the experimental trials. 41
27	Complete plan for Card part. 42
28	The complete successes, partial successes, and failures of the experiments. 44
29	The complete successes, partial successes, and failures of the experiments broken down by part. 44
30	The types of experimental failures broken down by part. Includes partial successes and complete failures. 45

CHAPTER I

INTRODUCTION

This work is an effort to join two existing areas of research: sensorless parts orienting and manipulation of deformable objects.

Sensorless parts orienting has proven quite successful in creating devices that are both flexible in the parts they can orient as well as ensuring with high probability that a part will be oriented. These devices are useful for assembly lines in factories that utilize automation or repetitive work that is done to identical parts.

The problem of sensorless parts orienting with a parallel-jaw gripper was first addressed by Goldberg [1] who combined prior work in compliant motion planning and the study of mechanical parts feeders. In this work, we utilize the analysis functions, *Diameter* and *Squeeze*, developed by Goldberg's expansion of Brost's work [2] on grasping with a parallel-jaw gripper. Brost's work was preceded by Mason [3] who was the first to analyze pushing in robot manipulation. Goldberg focused his work on non-deformable parts, and this work presents the results of a sensorless approach to a subset of deformable object manipulation, which is typically far from sensorless.

Goldberg's analysis [1] is founded on assumptions for rigid parts utilizing a pressure threshold to prevent deformation. Goldberg's method is not readily adaptable to deformable parts meaning there is a risk of damaging deformation, and his method creates a dependency when working with rigid parts. To address the possibility of damage to the part and overcome the need for a pressure threshold, we introduce a new algorithm and manipulator actions that will use a part's provided *geometric description* and *deformation model* to exploit the part's deformation and generate a

This thesis follows the style of *IEEE Transactions on Automatic Control*.

Plan comprised of the shortest sequence of manipulator actions guaranteed to orient the part up to symmetry in its polygonal convex-hull from any unknown initial orientation and pose. In addition to generating the plan, the algorithm estimates whether a given manipulator is sufficiently precise to perform the actions that guarantee the final orientation; this is dictated by the particular part geometry, *deformation model*, and the *manipulator action path planner* which contains *end-effector constraints* and any standard motion planner. Both the *deformation model* and *manipulator action path planner* are replaceable allowing for adjustments to the algorithm to accommodate various parts.

The complexity of successfully manipulating deformable objects with any level of dexterity is one that many have tried to overcome by incorporating the most precise sensors available or developing complex models with multiple parameters for increasing levels of customization. This adding of complexity is done in hopes of eliminating as much uncertainty as possible, but as Erdmann and Mason [4] noted, sensing allows recognition of action success rather than changes the actions used to solve a problem. Effectively, sensing changes the ordering of actions; therefore, the addition of sensors will not change what particular actions accomplish, rather additional sensors will increase the complexity of how the problem is solved.

The goal of sensorless parts orienting is to reduce the complexity of designing a new parts feeder, so it seems counter intuitive to replace one type of complexity with another. Just as sensorless manipulation is a response to the dependency on sensors that Mason [5] referred to as sensor abuse, this work adopts a similar philosophy when dealing with deformation and models of part deformation, i.e., it attempts to minimize the use of the model altogether recognizing that an exact characterization of how the part will deform is unrealizable in practice. In this work the model need only provide a region which bounds the part and for very few orientations, rather than

form a predictive model of precise deformation geometry which can depend on many complex parameters that need to be defined for the parts. Employing an algorithm which employs weaker information requirements means assuming less and ultimately results in an approach which requires less tuning, and therefore, is both more robust and general in practice. It is a case where employing an algorithm which employs weaker information requirements means assuming less and ultimately results in an approach which requires less tuning and is thus both more robust and general in practice.

The two ideas match in more than philosophy in this work: ambiguous configurations are resolved via a sequence of actions which exploit deformation of the part. With this in mind, we approach the problem of orienting deformable parts in a minimalist manner utilizing simple models and no sensors. We generate a sequence of actions that guarantees success from any initial configuration and the addition of sensors would only eliminate a few unnecessary actions for some of the initial configurations.

To show the validity of the algorithm we illustrate multiple examples of parts where orienting is achievable and examples of parts for which the precision of the manipulators is insufficient. We also show the success of our algorithm with the execution of 192 experimental trials of executable plans for six parts made of four types of materials: paper, textile, card, and foam.

Finally, we will discuss the experimental results in more detail along with future work, and the major contributions of this work:

1. This work is the first to address sensorless orienting of thin deformable parts.
2. The algorithm generates the shortest sequence of actions required to orient a deformable part up to symmetry in its convex-hull.

3. This work has applications of truly sensorless orientation of rigid (non-deformable) parts.

CHAPTER II

PREVIOUS WORKS

A. Sensorless Orienting

Sensorless parts orienting has proven quite successful in creating devices that are both flexible in the parts they can orient as well as ensure with high probability that a part will be oriented. Like this work, sensorless parts orienting grew from two other areas of research: compliant motion planning and the study of mechanical parts feeders. Compliant motion planning is founded on the idea of using mechanical compliance to reduce uncertainty, and mechanical parts feeders are devices specialized for repetitively orienting identical parts.

This work utilizes compliant motion analysis initially developed by Brost [2] in his study of grasp planning with uncertainty for a parallel-jaw gripper and then modified for sensorless parts orienting with a parallel-jaw gripper by Goldberg [1]. In his work [2] Brost developed a push diagram that represents all possible motions of a part grasped by a parallel-jaw gripper and then utilized it to develop a single step grasping plan robust to uncertainty in friction and orientation. Brost did this by using a rule found by Mason [3] for predicting the direction a part will rotate when pushed under the considerations of Coulomb friction.

Mason [3] was the first to analyze pushing in robot manipulation. Other implementations of orienting and posing with similar methods include: Akella and Mason's [6] positioning parts by pushing, Mani and Wilson's [7] pushing with a single planar fence, Peshkin and Sanderson's [8] study of fixed fences arranged over a moving conveyor belt and planning manipulations for objects that slide [9], and Erdmann and Mason's [4] analysis of orienting by tilting a rectangular tray allowing a part to

slide into the walls of the tray.

In [5], Mason points out that many roboticists today have reached a point of sensor dependency that approaches what he refers to as sensor abuse. He highlights the point that sensors are not the only means of obtaining knowledge and reducing uncertainty; other methods of dealing with uncertainty that are not fully dependent on sensors include: ignoring uncertainty, striving for actions that are less susceptible to uncertainty, and expectation. In this work we use these methods of dealing with the uncertainty resulting from a sensorless approach.

B. Manipulation of Deformable Objects

Many researchers have addressed the challenge of manipulating deformable objects creating a variety of methods that have been grouped together as robot manipulation of deformable objects. Henrich and Wörn [10] concluded after an overview of the state-of-the-art that successful manipulation of deformable parts relies on visual and tactile information, a multi-fingered gripper, two arm cooperation, and prior knowledge. It is important to note, Henrich and Wörn state sensors are necessary. The work presented here shows this not to be the case. The remaining components of successful manipulation are present in this work: two-fingered gripper, two robot manipulators, and prior knowledge is present in the understanding of the effects of the manipulator actions and the *deformation model*.

As shown by Henrich and Wörn, the manipulation of deformable objects is typically far from sensorless and can involve computation intensive modeling algorithms for the deformable object; therefore, it is not surprising that throughout the literature from some of the original works to the more recent we see a similar grouping of manipulators, sensors, and deformation model.

There are a variety of examples in the literature of applications of manipulation of deformable objects. Shepard et al. [11] worked to improve cloth grasping for improved robot towel folding. Hirai et al. [12] developed a control law that worked with a real-time vision system to translate, rotate, and grasp a deformable object. Still in the field of manipulating when there is deformation is the work of Lin and Lee [13] in which they consider the case when the manipulator is deformable.

These are only a sample of the many works focused on manipulation, and there are other works which are more concerned with the development of models that accurately describe deformation. Terzopoulos et al. [14] derived the underlying differential equations that model the elasticity of materials such as paper, rubber, cloth, and flexible metals allowing them to develop accurate graphical models.

Today, there are a variety of models for deformable bodies used in computer graphics. Gibson and Mirtich [15] surveyed the types of deformable modeling and noted that of the two types, geometric and physical, that the physical approach of finite element models offer the greatest accuracy, but is often too time intensive for graphics. In this work, accuracy is of greater importance, thus making finite element methods a good choice. For more information on finite element models refer to [16]. Another popular method is mass-spring models which represent a deformable body as a grid of points connected by springs [17], and within the geometric category of deformable body modeling there is the popular SPLINES [18, 19].

CHAPTER III

METHODS

A. The Parts

This work extends previous works [1, 2, 6] in which the dynamics of parts were analyzed using the parts' polygonal convex-hulls. We require that the parts are represented by their convex-hulls which must be simple polygons and are described by a list of vertices listed in a counter-clockwise fashion; we refer to this as the *geometric description*, Fig. 1. From the *geometric description*, edges are generated, and the first of these is used as the orientation vector. We will often refer to the orientation vector using o_i where $0 \leq o_i \leq 2\pi$.

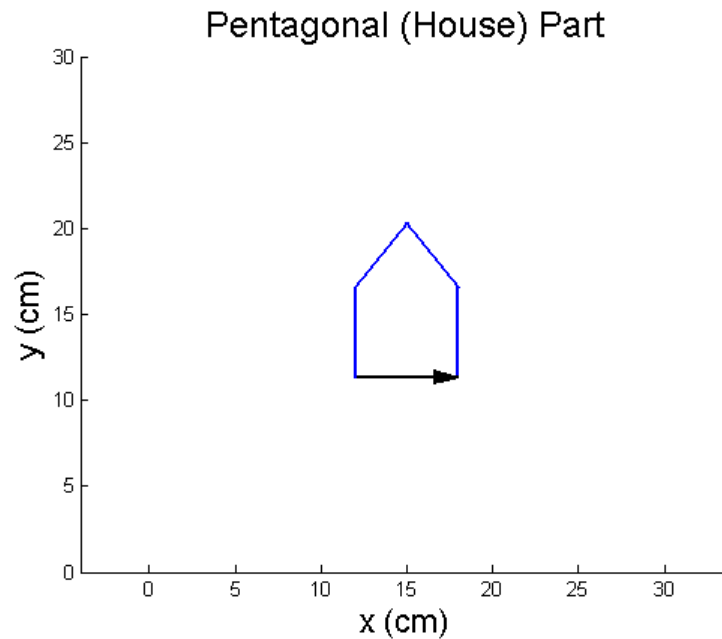


Fig. 1.: A part represented by the list of vertices of its convex-hull and orientation vector.

B. Goldberg's Functions

The functions developed by Goldberg [1] will be discussed in pairs. Each function takes a part in a particular orientation o_i . Within the pairs, one function will provide a distance measurement and the other function will provide the outcome orientation o_i following a particular action.

The first pair of functions are the $Diameter(o_i)$ and $Squeeze(o_i)$ functions (see Fig. 2). As the name suggests $Diameter(o_i)$ will provide the diameter of a part in the provided orientation. In [1], Goldberg defines the diameter as the distance separating the jaws of a parallel-jaw gripper when both jaws first make contact with the part. In this work there is a slight modification to this definition: the diameter is the distance between the sweeper (a movable wall) and its parallel fixed-wall when both walls make contact with the part. Next, $Squeeze(o_i)$ describes the rotational mappings of one orientation to another as the parallel-jaws or walls are moved closer together. $Squeeze$ received its name from the squeezing motion that occurs when a parallel-jaw gripper is closed. In this work we refer to this as sweeping rather than squeezing because a single movable wall resembles the motion of a broom towards the fixed-wall. Observing Fig. 2 we can see that when squeezed sufficiently [1], or swept, the part will rotate to one of the orientations at one of the minimums seen in the plot of $Diameter(o_i)$. These orientations are of great significance in this work and will be discussed in further detail in the Deformable Orientations section.

The second pair of functions is the $Radius(o_i)$ and $Push(o_i)$, see Fig. 3. $Radius(o_i)$ will determine the perpendicular distance from the contacting jaw of a parallel-jaw gripper (in this work the sweeper) to the center-of-mass (COM). The determined radius is utilized by $Push(o_i)$ to determine the rotational mapping of the part when it is pushed by a single jaw of a parallel-jaw gripper or by the sweeper.

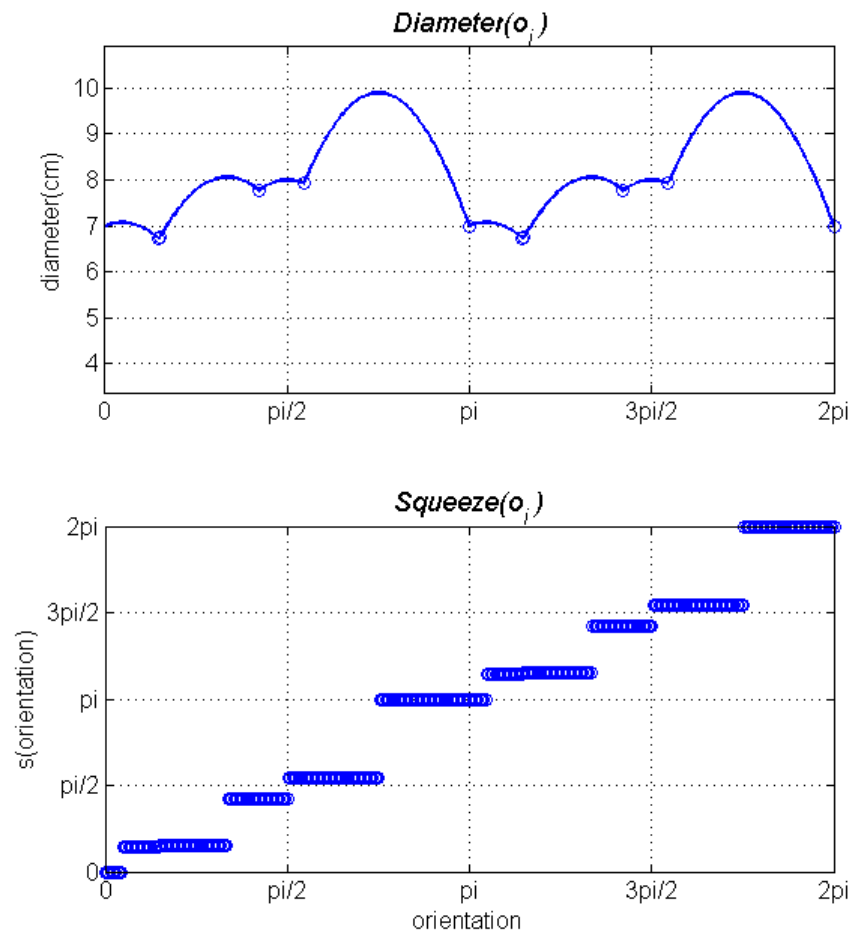


Fig. 2.: $Diameter(o_i)$ is a plot of the diameter for all orientations o_i . $Squeeze(o_i)$ is a plot of the final orientation after a sweep (squeeze) occurs, given an initial o_i .

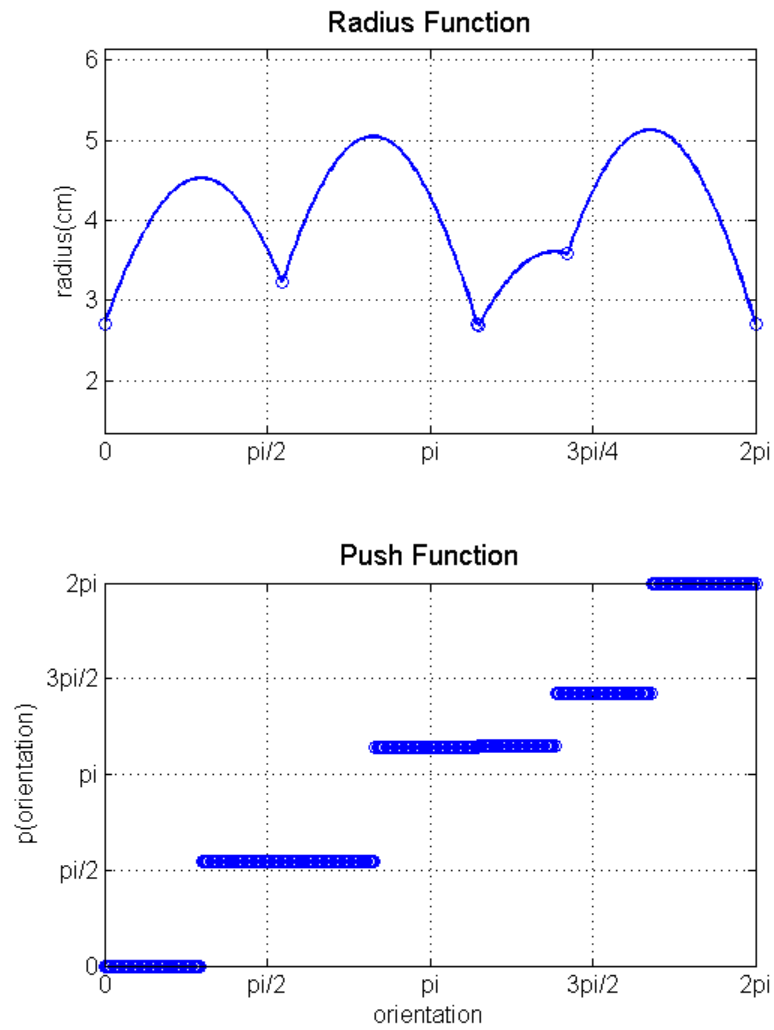


Fig. 3.: $Radius(o_i)$ is a plot of the radius for all orientations o_i . $Push(o_i)$ is a plot of the final orientation after a push occurs, given an initial o_i .

C. Assumptions

In this section we address the necessary assumptions for the applicability of this work. Assumptions 1-8 are from the works of Brost [2], Mason et al. [20], and Goldberg [1] with slight modifications to make them applicable to this work. Assumption 1 is a summarization of the assumptions of unwanted forces from previous works and the additional forces introduced by adding deformable parts. Assumption 3 has been modified to keep Goldberg's analysis functions, see Goldberg's Functions, applicable to the deformable parts. Assumption 9 is an addition that is necessary to describe the limitations of the deformation.

1. Inertial forces, frictional forces, and part elastic forces are negligible in comparison to forces applied by manipulators.
2. The sweeper and its fixed-wall can be modeled as a parallel-jaw; with all motions orthogonal to the jaws.
3. The convex hull of the part can be treated as a semi-rigid thin planar polygon.
4. Only one part is handled at a time.
5. The part's initial position is unconstrained as long as it lies somewhere between the walls. The part remains between the walls during sweeping, deforming, and pushing.
6. The part's center-of-mass (*COM*) is given and the coefficient of friction with the support surface is independent of position and velocity.
7. There are zero frictional forces between the walls and part.
8. Once contact is made between a wall and the part, the two surfaces remain in contact throughout the sweeping motion.
9. Elastic deformation occurs only in *stable orientations*. This is a limited set of few orientations, each of which must be modeled via a *deformation model*.

D. Deformable Orientations

Recall from the discussion of Goldberg's Functions that $Squeeze(o_i)$ and $Push(o_i)$ indicate that sufficient squeezes and pushes will reorient a part to one of the orientations at the minimums. These minimums form a set O of important orientations referred to as *stable orientations*; we refer to them as *deformable orientations*. Fig. 4 shows an example of a *deformable orientation* for a part referred to as a four-gon. As can be seen, a *deformable orientation* occurs when at least one edge aligns with the sweeper or its parallel fixed-wall. In this situation the rotational forces applied to the part sum to zero, preventing further rotation.

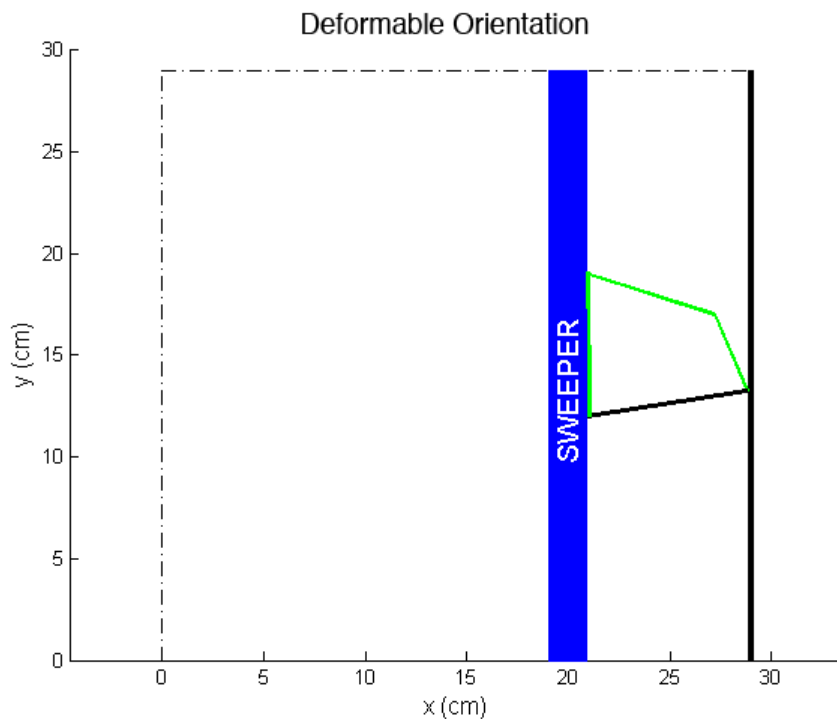


Fig. 4.: A *deformable orientation* of a four-gon part.

The significance of the *deformable orientations* to this work was first noted in Goldberg's requirement for his squeezing action to cease before deformation oc-

curred [1]. Under the assumptions, the *deformable orientations* are the only orientations in which deformation might occur if the necessary forces are applied.

Knowing the importance of the *deformable orientations*, we want to exploit these to allow orientating. Recall that we are only concerned with the squeeze portion of the sweeping action, and when we perform a sweep, it begins from a region of the workspace that the part is not present. Also notice that squeezes only take place once the part has been pushed into contact with the fixed-wall. This results in the first possible squeeze occurring at the orientation with the largest diameter, and results in a natural ordering of the *deformable orientations* from largest to smallest diameter, see Fig. 5.

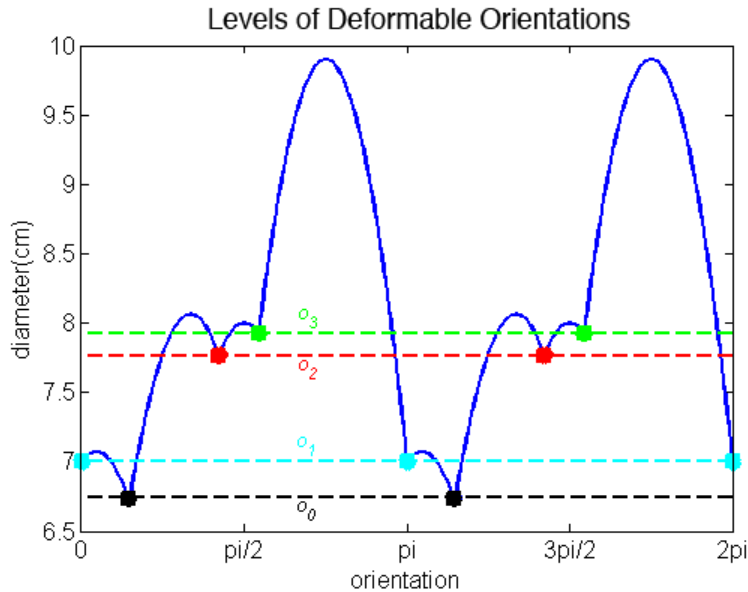


Fig. 5.: The k levels of *deformable orientations* ordered from smallest to largest diameter.

We exploit this ordering and add the symmetrical *deformable orientations* with the smallest diameter to the set of *terminating orientation*, o_0 . Therefore, the complete ordered set of k sets of *deformable orientations* is: $O = \{o_{k-1}, o_{k-2}, \dots, o_i, \dots, o_0\}$.

E. Adding Deformation

Ideally, we would like to be able to continue to squeeze until the part was successfully oriented to the o_0 ; however, this is not possible due to the existence of the *deformable orientations*. Goldberg [1] overcame the issue of *deformable orientations*, or *stable orientations* by preventing deformation via a pressure threshold and reorienting his parallel-jaw gripper. In this work, we do not allow the sweeper and its parallel fixed-wall to be reoriented; instead, we introduce new actions DEFORM, PUSH, GRASP, and REORIENT that allow us to exploit the deformation of the part.

In order to use these actions to exploit deformation we require three additions that we discuss in the following: the *deformation model*, the *end-effector constraints*, and the *manipulator action path planner*.

1. Deformation Model

The *deformation model* denoted as function $Deformation(o_i, \delta_i) \mapsto \langle Vol_i, Com_i \rangle$, is part specific, takes a given orientation o_i and a deformation distance δ_i for the part and provides a volume Vol_i that is guaranteed to contain the deformed part and a spherical volume Com_i guaranteed to contain the part's *COM*. The *deformation model* must also return an indication that a part will be damaged when δ_i is too great (formally, this can be treated as returning a zero volume).

Hirai et al. [21] worked on modeling thin deformable parts, similar to those presented in this work, by analyzing the potential energy under geometric constraints. A model of this type would be more accurate, but we will show that this level of accuracy is not always necessary and that a *deformation model* of choice can be used.

As illustrated in Fig. 6, the tent model used in the experiments follows the definition of the *deformation model* and provides Vol and Com by modeling the

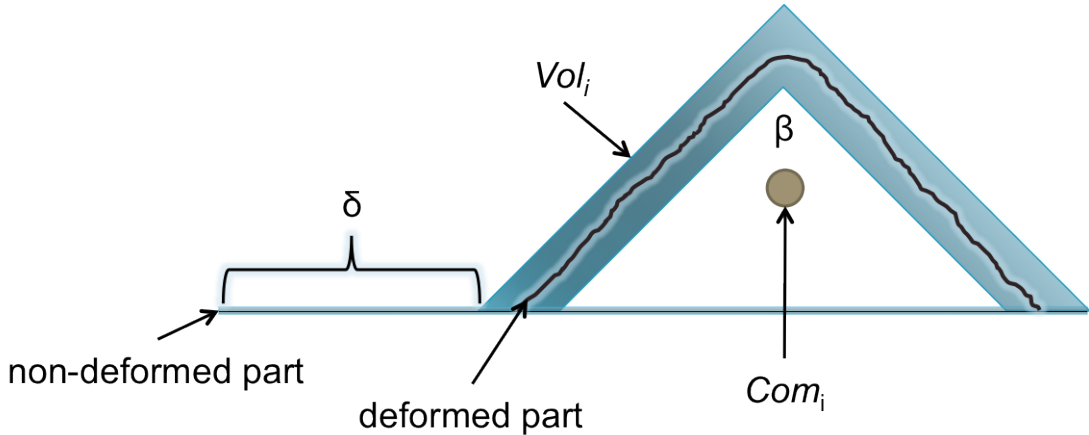


Fig. 6.: A linear *deformation model* for an arbitrary part; referred to as a tent model. δ is a deformation distance. β is the angle used to describe excessive deformation. Vol_i is a volume created by incorporating tolerances of the part deformation and is guaranteed to contain the deformed part. Com_i is a volume guaranteed to contain the part's *COM*.

deformation with a tent or hinge model. Recall the Vol and Com returned are volumes guaranteed to contain the deformed part. Due to the simplicity of the tent model, these volumes can be inaccurate (overly large) if tolerances are large. Safe deformation is determined by the angle β which is a minimum of the tent peak angle that must not be exceeded. When β is exceeded, then the tent model will return an indication of unsafe deformation.

2. End-effector Constraints

When considering the actions PUSH, GRASP, and REORIENT it was necessary to incorporate a motion planner for the manipulator. We did not want to constrain this work to any specific motion planner. Thus, we developed *end-effector constraints* that can use by any typical motion planner to guarantee successful PUSH, GRASP, and RE-

ORIENT actions utilizing the manipulators. There are other styles of manipulators and grippers that would benefit from a redefinition of these constraints and these are discussed briefly in the Conclusions.

a. Push Constraints

The first *end-effector constraints* are for PUSH, see Fig. 7. These constraints guarantee a successful PUSH. These are volumetric constraints that guarantee a successful PUSH. For a PUSH we utilize the following volumetric constraints to assure success. The *deformation model* will return a volume guaranteed to contain the deformed part and a volume guaranteed to contain the deformed part's *COM*. Utilizing these volumes we need the *manipulator action path planner* to generate a \tilde{P}_{path} that will push the part without applying enough rotational forces to topple the part. We begin by projecting a volume, less the manipulator positioning tolerances, along the normal of the gripper. This volume must contain the volume containing the *COM*. Next, we project a similar volume along the potential \tilde{P}_{path} that must contain a push-contact surface area on the part. The motion planner is then free to generate a sequence of collision free configurations that meet these constraints and will push the part to the corner. For the final $P_0 = \text{PUSH}(\tilde{P}_{path-0})$ the constraints are loosened to collision free pushing with the gripper tip pushing along the work surface.

b. Grasp Constraints

For the GRASP *end-effector constraints* we followed the work of Smith et al. [22], see Fig. 8. For a more accurate GRASP applicable to more manipulators the work of Gopalakrishnan and Goldberg [23] should be considered. In their work, Gopalakrishnan and Goldberg expanded rigid part grasping to deformable parts taking into consideration factors that included deformation due to grasping. Smith et al. [22]

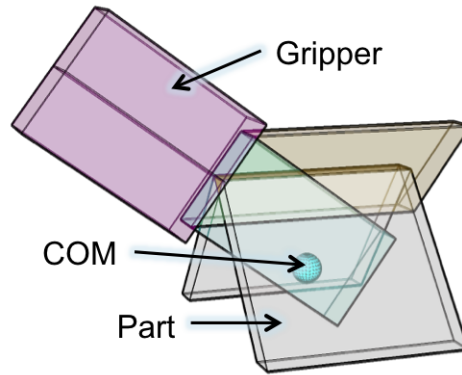


Fig. 7.: The PUSH *end-effector constraints*.

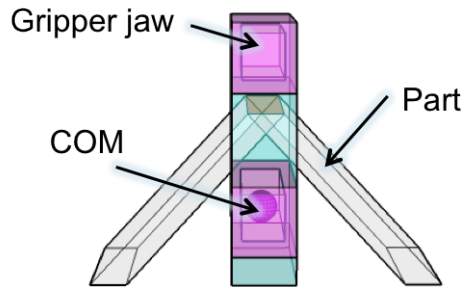


Fig. 8.: The GRASP *end-effector constraints*.

computed parallel-jaw grips for rigid polygonal parts by placing two grip points representing the jaws of the gripper collision free around the part. The placement of the grip points is constrained by the grip axis, the line joining the grip points, which must pass through the part's center-of-mass. We modify this last requirement as follows: the grip axis is extended beyond both grip points and a volume based on the dimensions of the jaws is projected along the grip axis. This grip volume must contain the Com_i provided by the *deformation model*. We assume that the foam lining the parallel-jaw gripper prevents undesirable deformation of the part during grasping.

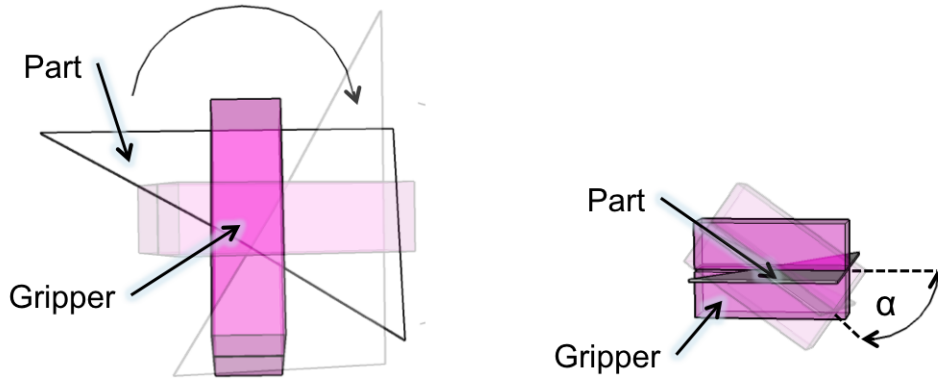


Fig. 9.: The REORIENT *end-effector constraints*.

c. Reorient Constraints

REORIENT is a complex action akin to a pick and place operation that we have strived to define by a simple set of *end-effector constraints*, see Fig. 9. Utilizing an arbitrary motion planner we place the following requirements on the manipulator configuration path. From the final GRASP manipulator configuration there must exist a manipulator configuration that holds the part at $o_0 \pm \phi$ with the parallel-jaws at an angle α for which the gravitational forces are sufficient to pull the part to the work-surface. This assumes that the release/place phase of REORIENT is smooth and slow enough that the part will stay in $o_0 \pm \phi$.

The tolerance ϕ is extracted from the *Diameter* plot and is the maximum allowed reorientation variance from the goal orientation o_0 that will still guarantee successful reorientation.

For example, consider the four-gon part with $\phi = \frac{\pi}{10}$, see Fig. 10, and the manipulators which are capable of a $\frac{\pi}{2} \pm \frac{\pi}{30}$ reorientation. This combination indicates there is sufficient precision for REORIENT to duplicate the mappings shown in Fig. 11.

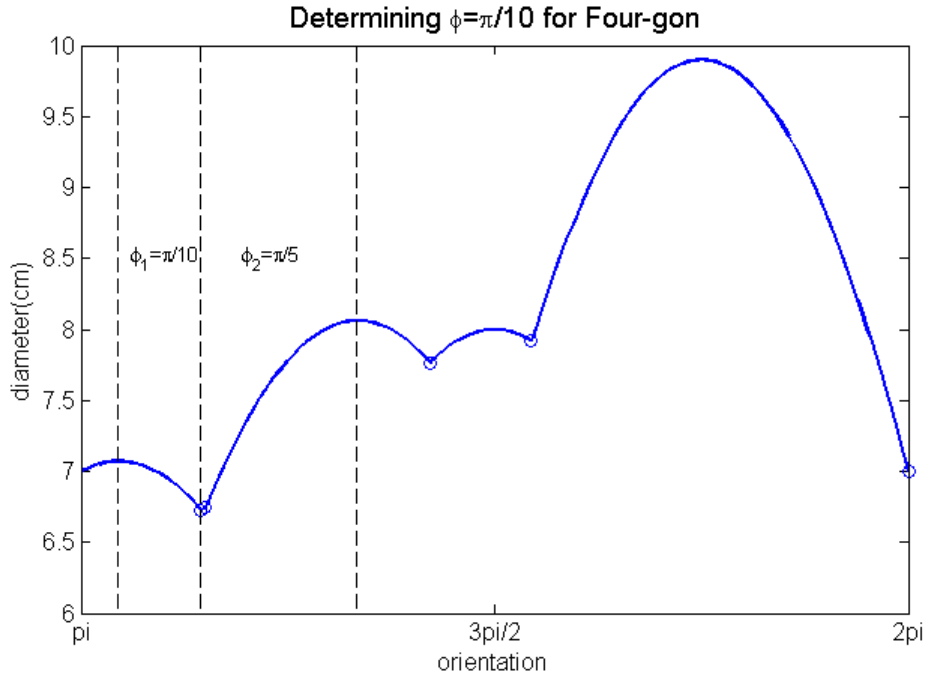


Fig. 10.: Example of reorientation tolerance ϕ for a four-gon part.

3. Manipulator Action Path Planner

When provided with the set of symmetrical *deformable orientations* O , *deformation model*, and the *end-effector constraints*, the *manipulator action path planner* can then generate a push path \tilde{P}_{path} , grasp path \tilde{G}_{path} , and reorient path \tilde{R}_{path} , and determine a safe and sufficient δ . The generation of the action paths is achieved by querying the *deformation model* with *deformable orientations* and possible δ then utilizing the returned Vol_i and Com_i along with a motion planner and the *end-effector constraints*. A safe and sufficient δ is found during the querying process by an increasing counter.

At this point, this work is solely focused on orienting deformable parts, but in the Conclusions, we suggest additions that would allow orienting of both rigid and deformable parts.

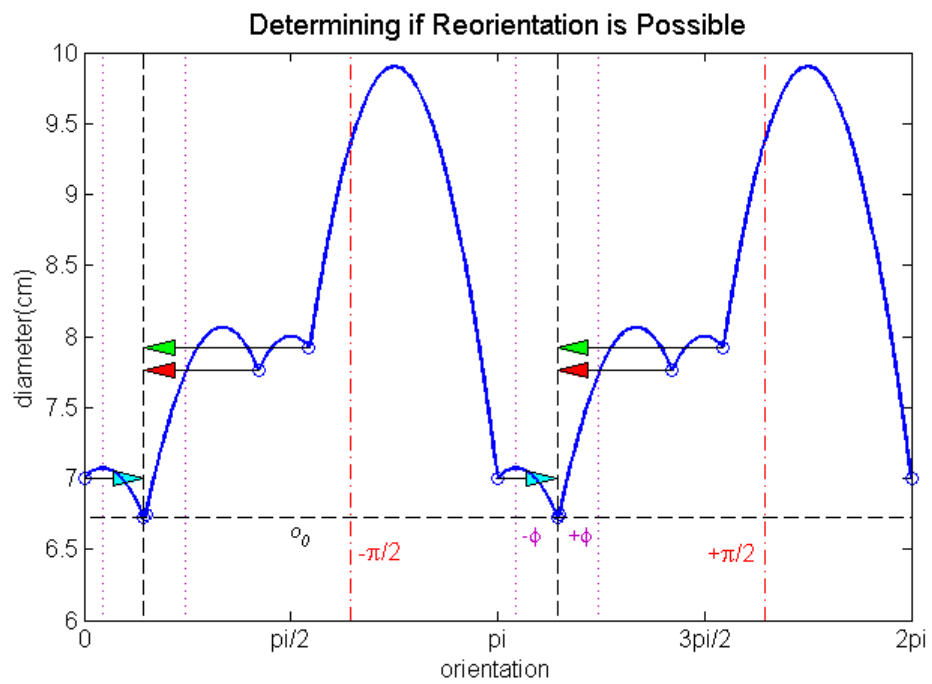


Fig. 11.: Determining if the manipulator precision is sufficient for reorientation of part o_0 . Fig. 10 showed $\phi = \frac{\pi}{10}$. Manipulator is capable of $\frac{\pi}{2} \pm \frac{\pi}{30}$ REORIENT.

F. Manipulator Actions

Now that the methods of analysis have been explained we can describe the robot manipulator actions and how they map o_i to o_0 .

To begin we will describe $\text{SWEEP}(Diameter(o_i))$, see Fig. 12, which when provided with an orientation o_i of the part will sweep up to the diameter of o_i . SWEEP is modeled after the SQUEEZE action in [1], but rather than using a parallel-jaw gripper we use a sweeper (a moveable wall) in combination with a parallel fixed-wall.

Goldberg [1] generated plans consisting of SQUEEZE s and reorients of the parallel-jaw gripper. In this work, we do not allow the sweeper and its parallel fixed-wall to be reoriented; instead, we introduce new actions DEFORM, PUSH, GRASP, and REORIENT that allow us to exploit the deformation of the part.

Recall that SWEEP has two phases: push and squeeze, and that the push phase will map parts towards the wall until they enter the squeeze phase.

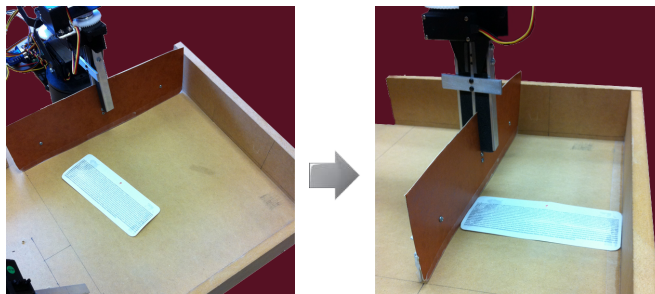


Fig. 12.: SWEEP: a primary robot action for reducing possible part poses and orientations.

This mapping allows us to be concerned only with the analysis of the squeeze phase shown in Fig. 13.

Ideally, a single SWEEP action would be sufficient to successfully reorient the part to o_0 , but this is not the case due to the *deformable orientations* which cease to

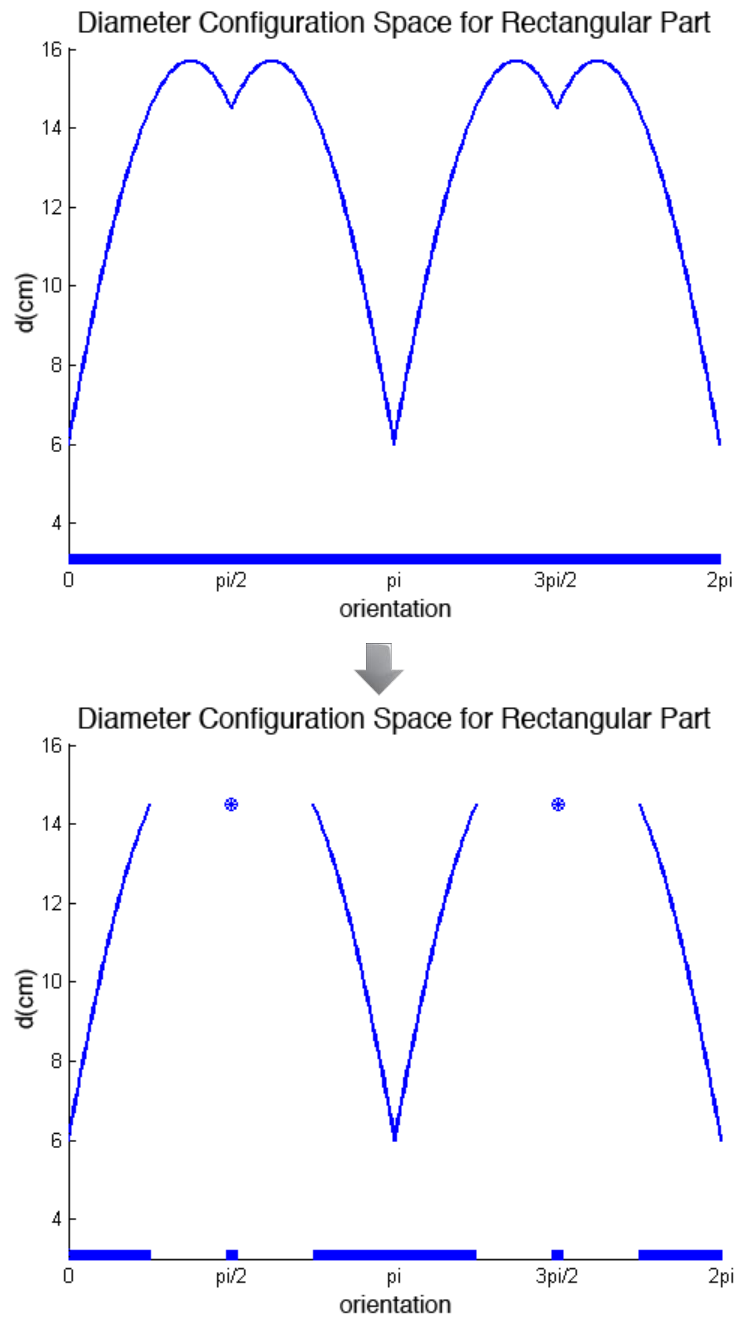


Fig. 13.: Example mappings of SWEEP on possible part orientations of a rectangular part. Shows how parts are reoriented to the *deformable orientations*, which are at the minima in the *Diameter* function.

rotate and begin to deform under additional sweeping motions.

To eliminate this issue we introduce the remaining actions starting with the $\text{DEFORM}(Diameter(o) - \delta)$ action, Fig. 14. DEFORM , similar to SWEEP , is a sweeping motion that is applied only a distance of δ (recall, δ is the sweeping distance for safe and sufficient deformation) beyond the previous SWEEP to possibly deform the part.



Fig. 14.: DEFORM : a primary robot action, a more precise version of SWEEP .

When the part is not in the orientation o_i , DEFORM is a small SWEEP and achieves the same mappings, but when the part is in o_i then DEFORM maps as shown in Fig. 15.

In this work, we have only guaranteed a final orientation, but with sweeping motions of SWEEP and DEFORM as well as the pushing motions of $\text{PUSH}(\tilde{P}_{path})$ we also achieve positioning of the part. Previously we discussed Goldberg's *Push* function, now we introduce an unrelated manipulator action PUSH . $\text{PUSH}(\tilde{P}_{path})$ will call the secondary manipulator to follow \tilde{P}_{path} pushing a deformed part and reducing the possible positions for grasping, see Fig. 16 and Fig. 17.

Once a part has been deformed and pushed $\text{GRASP}(\tilde{G}_{path})$, see Fig. 18, and $\text{REORIENT}(\tilde{R}_{path})$, see Fig. 19 actions can be performed to map the part to within ϕ of o_0 (illustrated in Fig. 20). See section End-effector Constraints for details on the orientation tolerance ϕ .

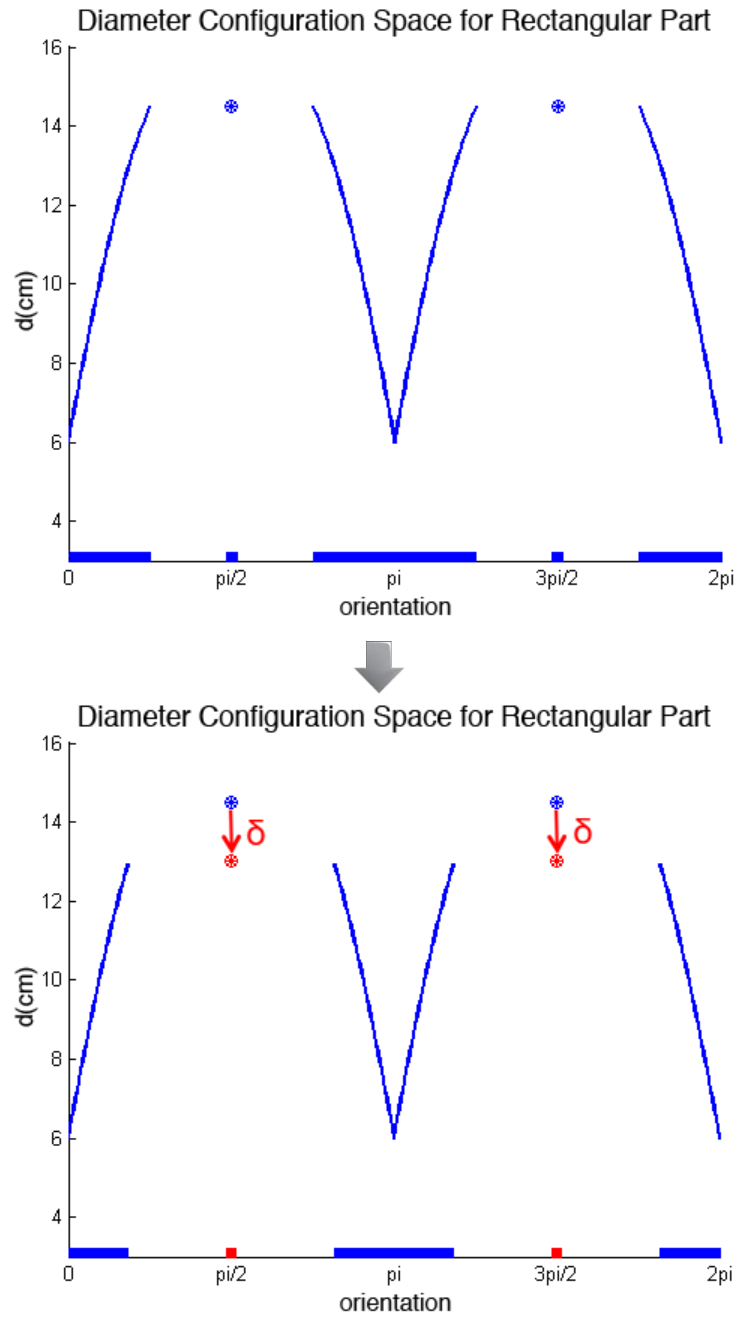


Fig. 15.: Example mappings of DEFORM on possible part orientations of a rectangular part. Shows how a DEFORM is a small SWEEP that reorients parts not in *deformable orientations* and deforms parts by δ_i in *deformable orientation* o_i .

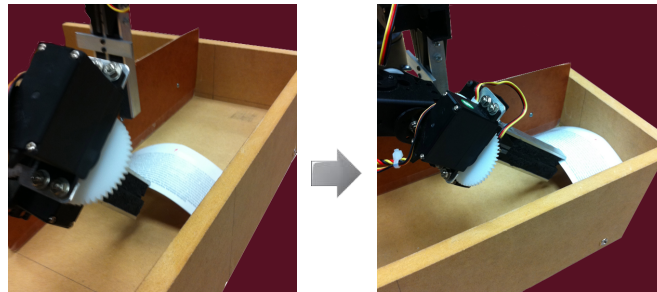


Fig. 16.: PUSH is a secondary robot action that reduces possible deformed part poses for GRASP.

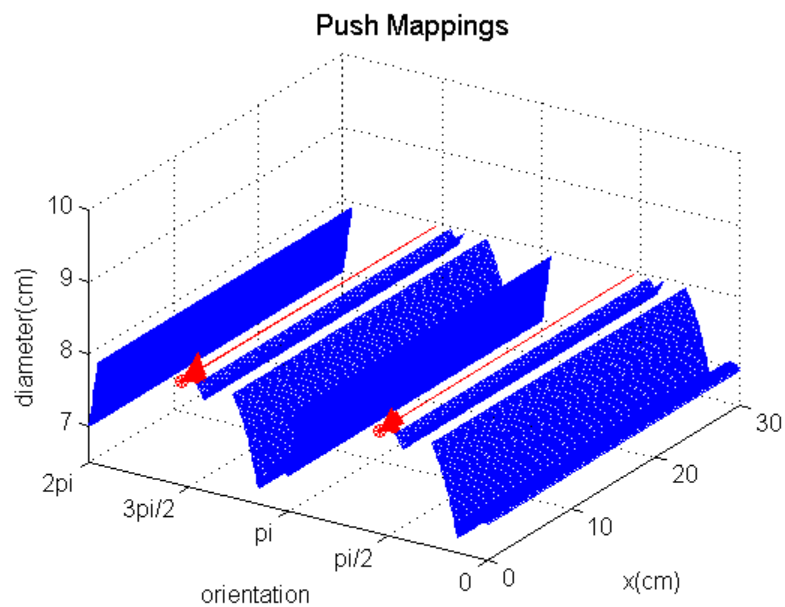


Fig. 17.: PUSH maps the deformed part to the corner position for grasping. A deformed part in o_i will be at some $\langle X, y_i \rangle$ where $X = [0, 29]$. PUSH will reduce this to $\langle x_0, y_i \rangle$ where x_0 is the corner position. Mappings shown are for four-gon part.

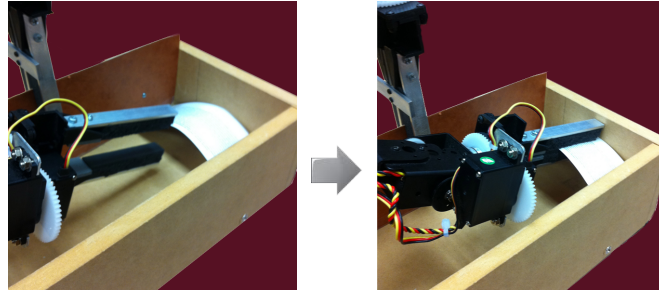


Fig. 18.: GRASP: a secondary robot action that precedes REORIENT.

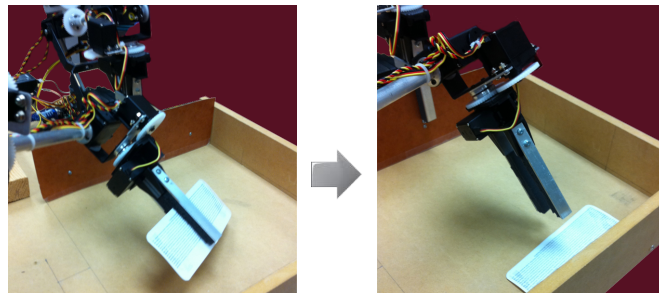


Fig. 19.: Reorient: a secondary robot action for mapping deforming orientations to within ϕ of the *terminating orientation* o_0 .

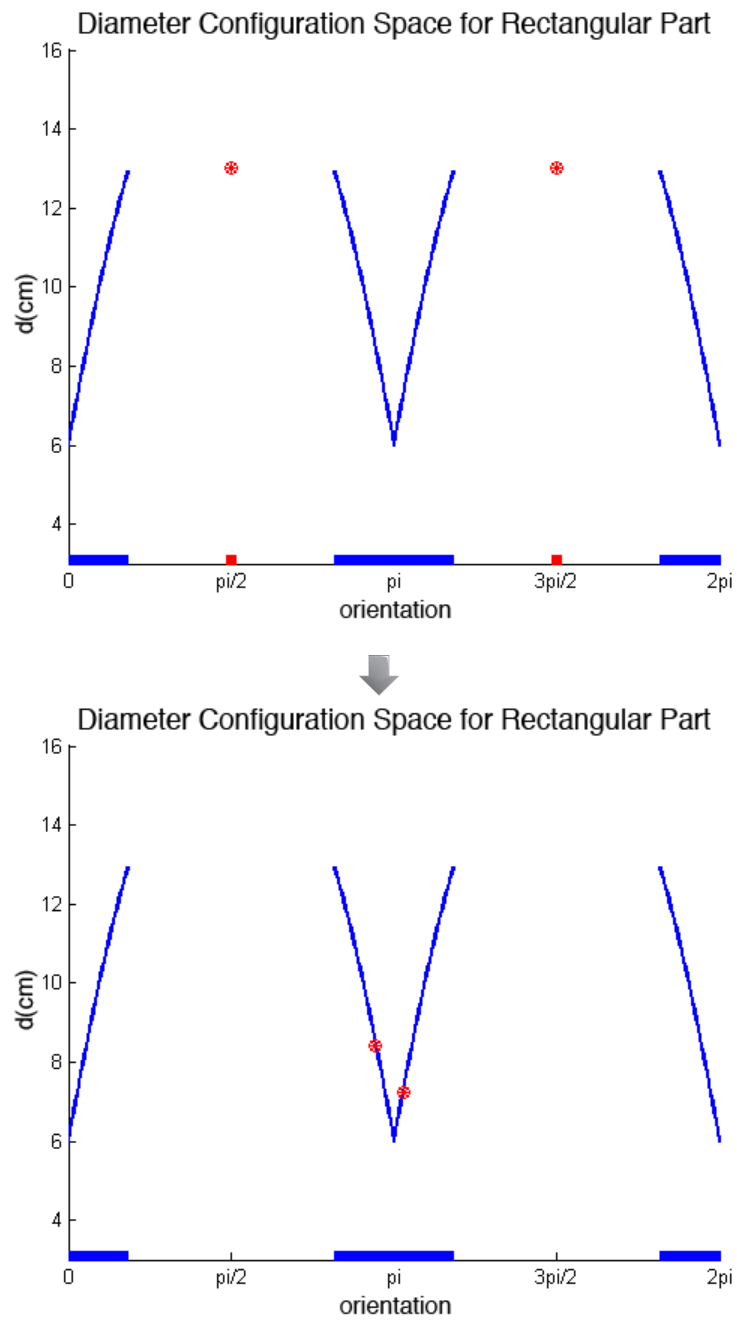


Fig. 20.: Example mappings of a GRASP followed by REORIENT on the possible orientations of a rectangular part. Shows how a deformed part is mapped to $\theta_0 \pm \phi$.

G. Algorithm

In this section we will discuss in detail the algorithm, refer to Algorithm 1. We begin by requiring input for a particular part: the *geometric description* and the *deformation model*. These are described in more detail in the sections: The Part and Adding Deformation.

Once the algorithm has been provided with the *geometric description* and a *deformation model* it will find the set O of *deformable orientations*.

This is accomplished by first generating the *Diameter* function and then applying the *Squeeze* function described in the section Goldberg's Functions. The application of the *Squeeze* function identifies the *deformable orientations*, which we then group into sets based on symmetry in *Diameter* and place in O . The next step is to order the set O from smallest to largest diameter.

Working in reverse order of the ordered set of *deformable orientations* O , the algorithm will first call upon the *manipulator action path planner* to determine a safe and sufficient δ_i and calculate \tilde{P}_{path-i} , \tilde{G}_{path-i} , and \tilde{R}_{path-i} .

To determine δ_i and the action paths the *manipulator action path planner* shown in Algorithm 2, requires the symmetrical subset of *deformable orientations* o_i and the *deformation model*. The *manipulator action path planner* begins by initializing Δ a user specified parameter for the step size of δ_i , each of the action constraints, the work space configuration WS , a container VC for the volumes returned by the *deformation model*, δ_i , and the action paths.

The *manipulator action path planner* will pass WS , VC , and the relevant action constraints to the motion planner to determine each of the action paths. For the *terminating orientation* the *manipulator action path planner* will only find a \tilde{P}_{path-0} , and for any other o_i the *manipulator action path planner* will determine a δ_i while

querying the motion planner for a \tilde{P}_{path-i} , \tilde{G}_{path-i} , and \tilde{R}_{path-i} .

Once the *manipulator action path planner* has returned the action paths and δ_i , the algorithm will determine if the precision is sufficient by assuring that action paths were found and that a $\text{DEFORM}(\text{Diameter}(o_i) - \delta_i)$ will not deform o_{i-1} . This check of δ_i is necessary to maintain the ability to distinguish between o_i and o_{i-1} . For the *terminating orientation*, the algorithm only needs to assure \tilde{P}_{path-0} exists to check for sufficient precision.

There are three reasons the algorithm might determine there is insufficient precision:

1. The *deformation model* is too pessimistic, i.e. volumes are too large.
2. Robots are too imprecise, so the tolerances are too large and the *end-effector constraints* are unable to be met.
3. The motion planner is not complete.

If the precision is insufficient then the algorithm will return an indication that greater precision or a more complete motion planner is needed; otherwise, when precision is sufficient, the algorithm will generate a sequence of $S_i = \text{SWEEP}(\text{Diameter}(o_i))$, $D_i = \text{DEFORM}(\text{Diameter}(o_i) - \delta_i)$, $P_i = \text{PUSH}(\tilde{P}_{path-i})$, $G_i = \text{GRASP}(\tilde{G}_{path-i})$, and $R_i = \text{REORIENT}(\tilde{R}_{path-i})$ for each *deformable orientation* with a diameter greater than the *terminating orientation* o_0 terminated by a sequence of $S_0 = \text{SWEEP}(\text{Diameter}(o_0))$, and $P_0 = \text{PUSH}(\tilde{P}_{path-0})$ for the *terminating orientation* o_0 . This combined sequence of actions will be returned as the *Plan* to be executed by the robot manipulators.

Algorithm 1 The Algorithm

Input: *Geometric description:* a counter-clockwise list of vertices of the convex-hull:

$$geo = \{\langle x_0, y_0 \rangle, \langle x_1, y_1 \rangle, \dots, \langle x_k, y_k \rangle\}$$

Deformation model:

$$DM(o_i, \delta_i) \mapsto \langle Vol_i, Com_i \rangle$$

- 1: Find the k *deformable orientations* O , by utilizing Goldberg’s *Squeeze* [1] and grouping symmetrical *deformable orientations* using Goldberg’s *Diameter*.
- 2: Order O such that $Diameter(o_0) < \dots < Diameter(o_{k-1})$
- 3: **for** $i = k - 1$ down to 1 **do**
- 4: $\langle \tilde{P}_{path-i}, \tilde{G}_{path-i}, \tilde{R}_{path-i} \rangle \leftarrow MAPP(o_0, DM(o_i, \delta_i))$
- 5: **if** $((\tilde{P}_{path-i} = \emptyset \vee \tilde{G}_{path-i} = \emptyset \vee \tilde{R}_{path-i} = \emptyset) \vee (Diameter(o_i) - \delta_i < Diameter(o_{i-1})))$
 then
- 6: **return:** “Insufficient precision for o_i or motion planner incomplete.”
- 7: **end if**
- 8: $S_i = \text{SWEEP}(Diameter(o_i))$
- 9: $D_i = \text{DEFORM}(Diameter(o_i) - \delta_i)$
- 10: $P_i = \text{PUSH}(\tilde{P}_{path-i})$
- 11: $G_i = \text{GRASP}(\tilde{G}_{path-i})$
- 12: $R_i = \text{REORIENT}(\tilde{R}_{path-i})$
- 13: **end for**
- 14: $\tilde{P}_{path-0} \leftarrow MAPP(o_0, DM(o_0, \delta_0 = 0))$
- 15: **if** $(\tilde{P}_{path-0} = \emptyset)$ **then**
- 16: **return:** “Insufficient precision for o_0 or motion planner incomplete.”
- 17: **end if**
- 18: $S_0 = \text{SWEEP}(Diameter(o_0))$
- 19: $P_0 = \text{PUSH}(\tilde{P}_{path-0})$

Output: $Plan = \{S_{k-1}, D_{k-1}, P_{k-1}, G_{k-1}, R_{k-1}, \dots, S_0, P_0\}$

Algorithm 2 Manipulator Action Path Planner

Input: Part orientation: o_i

Deformation model:

$$DM(o_i, \delta_i) \mapsto \langle Vol_i, Com_i \rangle$$

```

1:  $\Delta$ : user specified parameter
2:  $PC_i \leftarrow PUSH(o_i)_{constraints}$  {PUSH end-effector constraints for all  $o_i > o_0$ }
3:  $PC_0 \leftarrow PUSH(o_0)_{constraints}$  {PUSH end-effector constraints for  $o_0$ }
4:  $GC \leftarrow GRASP_{constraints}$  {GRASP end-effector constraints for all  $o_i > o_0$ }
5:  $RC \leftarrow REORIENT_{constraints}$  {REORIENT end-effector constraints for all  $o_i > o_0$ }
6:  $WS \leftarrow$  work space configuration following previously determined actions
7:  $VC \leftarrow \emptyset$  {container for deformation model volumes}
8:  $\delta_i \leftarrow 0$ 
9:  $\tilde{P}_{path-i} \leftarrow \tilde{G}_{path-i} \leftarrow \tilde{R}_{path-i} \leftarrow \emptyset$ 
10: if  $o_i = o_0$  then
11:   for each  $o_j$  in  $o_i$  do
12:      $VC \leftarrow VC \cup DM(o_j, \delta_i)$ 
13:   end for
14:    $\tilde{P}_{path-i} \leftarrow MotionPlanner(WS, PC_0, VC)$ 
15: else
16:   for  $\delta_i = 0$  to  $Diameter(o_i)$  do
17:      $\delta_i \leftarrow \delta_i + \Delta$ 
18:     for each  $o_j$  in  $o_i$  do
19:        $VC \leftarrow VC \cup DM(o_j, \delta_i = 0)$ 
20:     end for
21:      $\tilde{P}_{path-i} \leftarrow MotionPlanner(WS, PC_i, VC)$ 
22:      $\tilde{G}_{path-i} \leftarrow MotionPlanner(WS, GC, VC)$ 
23:      $\tilde{R}_{path-i} \leftarrow MotionPlanner(WS, RC, VC)$ 
24:     if  $(\tilde{P}_{path-i} \neq \emptyset) \wedge (\tilde{G}_{path-i} \neq \emptyset) \wedge (\tilde{R}_{path-i} \neq \emptyset)$  then
25:       return:  $\langle \tilde{P}_{path-i}, \tilde{G}_{path-i}, \tilde{R}_{path-i}, \delta_i \rangle$ 
26:     end if
27:   end for
28: end if

```

Output: $\langle \tilde{P}_{path-i}, \tilde{G}_{path-i}, \tilde{R}_{path-i}, \delta_i \rangle$

CHAPTER IV

VALIDATION

A. Examples

This section provides both types of examples: parts for which a plan executed on our robots can guarantee successful orienting of the part, and examples of parts that require robots with greater precision than those we employed.

1. Parts with Executable Plans

The first example is a part with a square convex-hull, see Fig. 21. Since the part has edges of equal lengths, this square part is fully symmetrical and has $k = 1$. This results in a plan consisting of only one SWEEP, and PUSH to orient and position the part up to symmetry.

The second example is a part with a right triangle convex-hull, see Fig. 22. This part has only two edges with unique lengths which is reflected by $k = 2$. The plan for this part is: $\{S_1, D_1, P_1, G_1, R_1, S_0, P_0\}$, where $S_1 = \text{SWEEP}$ parameterized for the set of symmetrical orientations o_1 . For this triangular part $o_1 = \{0, \pi, 2\pi\}$.

The third example, Fig. 23, takes a look at a part with a pentagonal convex-hull that Goldberg used in his examples and referred to as a house part. Again, due to symmetry this part has only three non-symmetrical orientations, $k = 3$. The plan for this pentagonal part is: $\{S_2, D_2, P_2, G_2, R_2, S_1, D_1, P_1, G_1, R_1, S_0, P_0\}$.

2. Parts Requiring Greater Precision

Now we will present a triangular part, Fig. 24a for which the algorithm outputs “Insufficient precision.” Referring to Fig. 24b, we can see that the triangular part

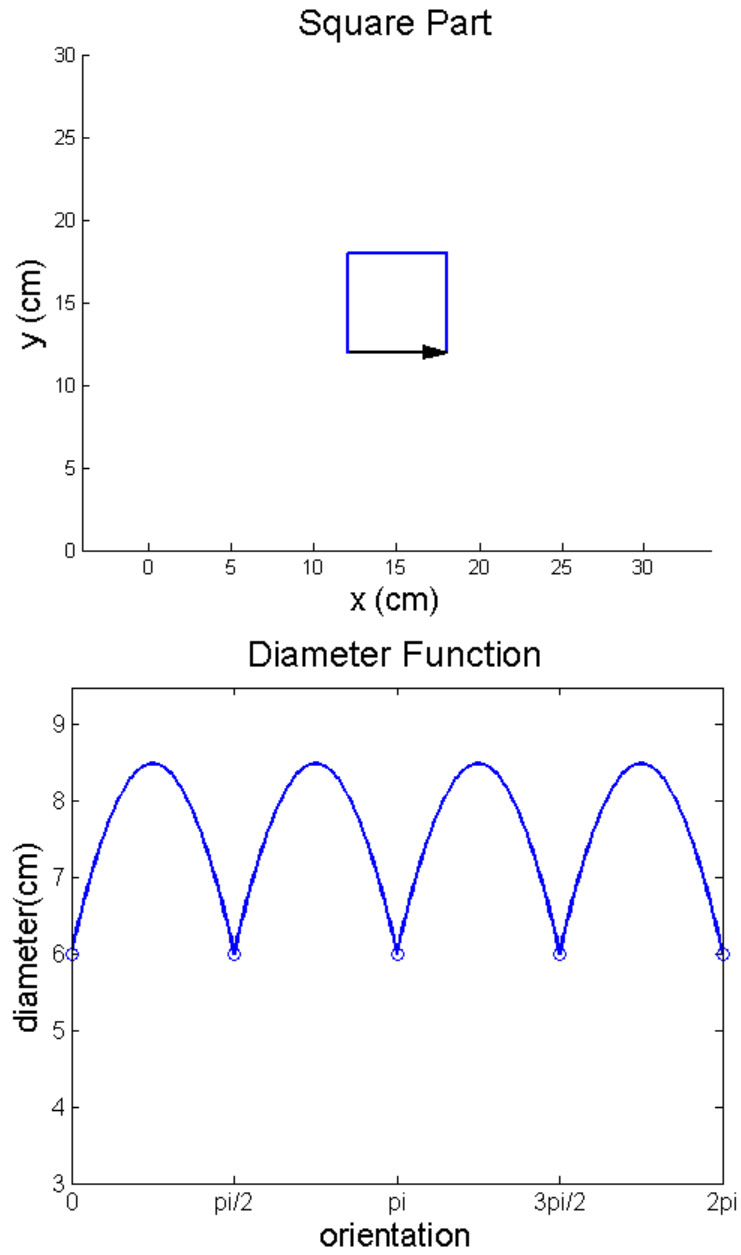


Fig. 21.: A square part which due to symmetry has $k = 1$.

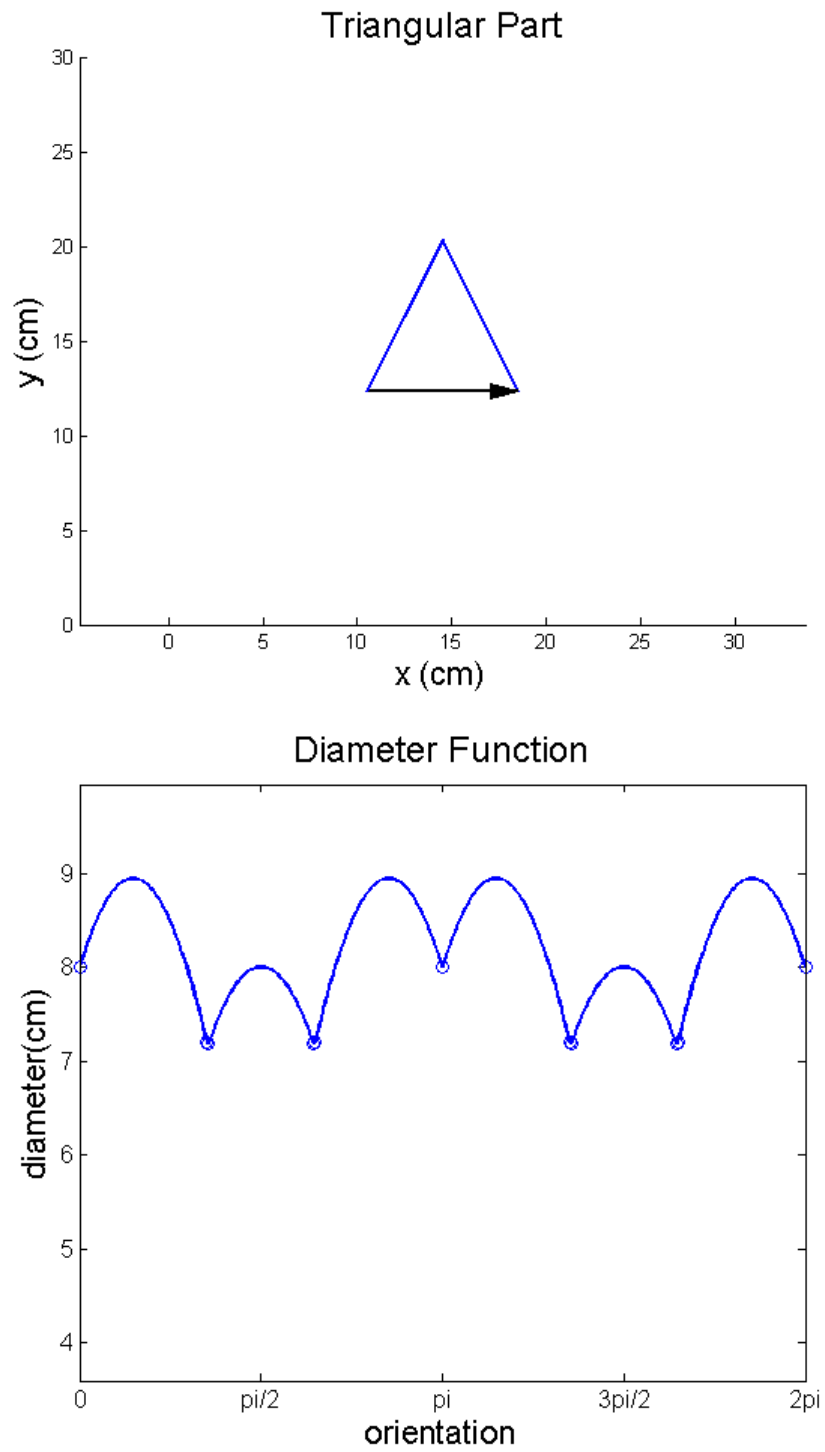


Fig. 22.: A triangular part which has $k = 2$.

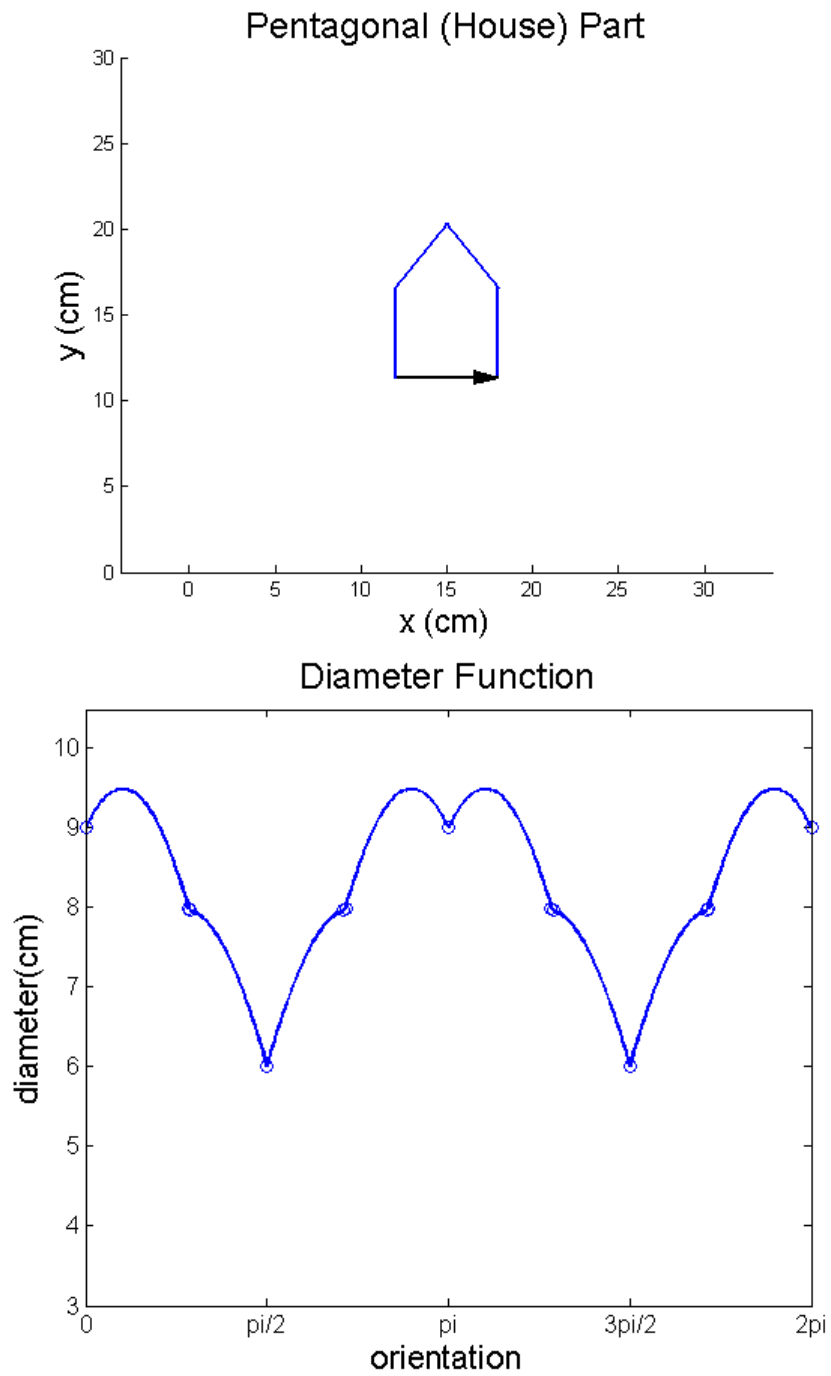


Fig. 23.: A pentagonal (house) part which has $k = 3$.

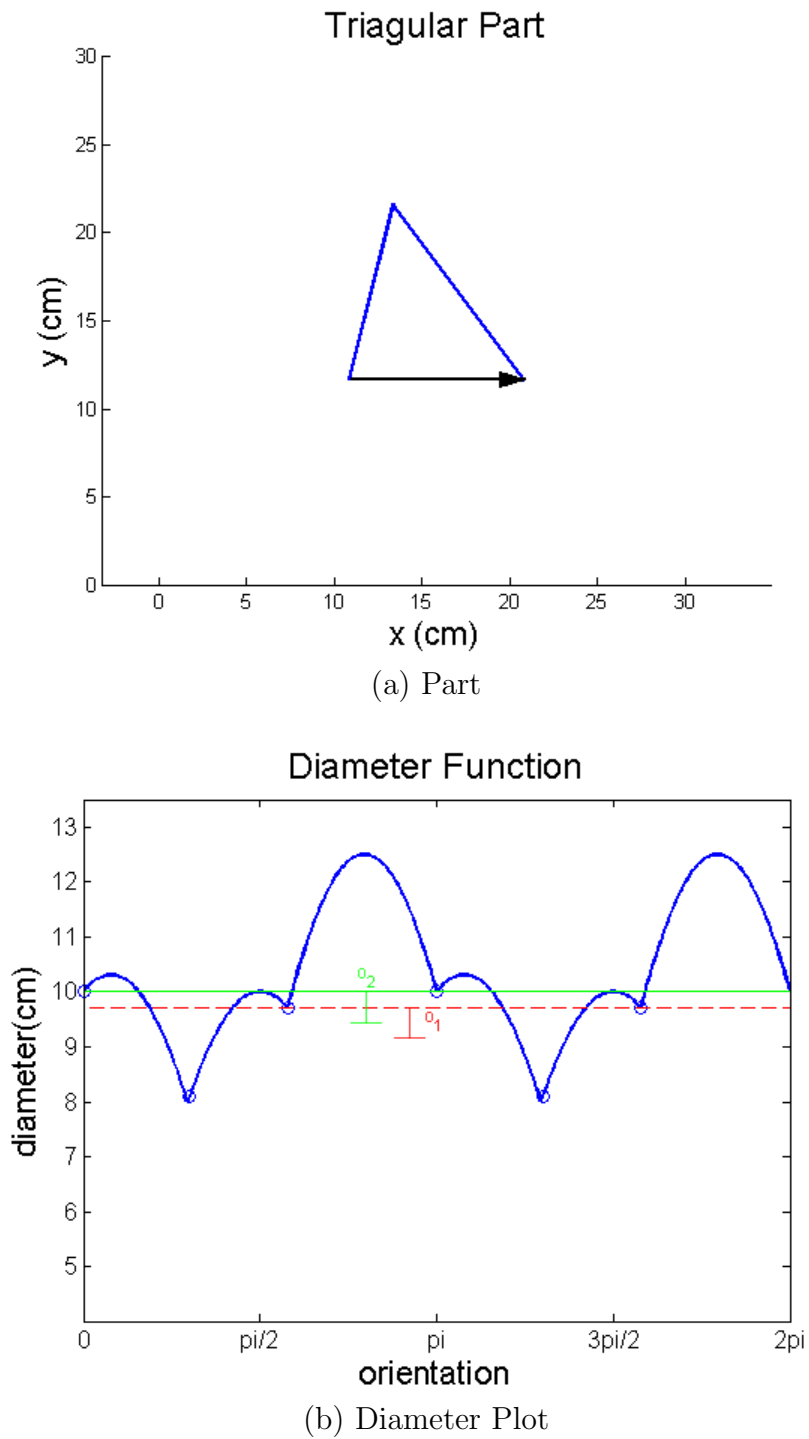


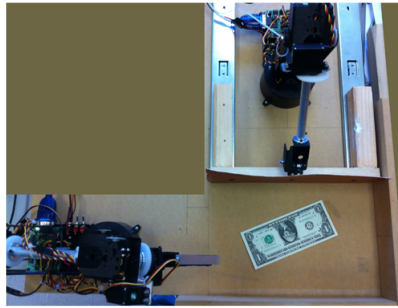
Fig. 24.: A triangular part for which the manipulators have insufficient precision due to overlapping deltas.

has $k = 3$ and that o_2 at *diameter* = 10.0cm and o_1 at *diameter* = 9.7cm are only separated by 0.3cm. Due to the tolerances of the robots, the *manipulator action path planner* utilizing the *deformation model* determined a safe and sufficient $\delta_2 = 1.0cm$. Applying a $\text{DEFORM}(\delta_2)$ would reduce the *diameter* of the sweeper and its fixed wall to 9.0cm which would result in deformation of a part in o_1 . The possibility of both o_2 and o_1 deforming is unacceptable for two reasons. The first, is that the ability to distinguish between o_2 and o_1 is lost because it is now possible for $\text{PUSH}(\tilde{P}_{path-2})$ and $\text{GRASP}(\tilde{G}_{path-2})$ to interact with both o_2 and o_1 rather than just o_2 . The second, is the potential for o_1 to be deformed more than its safe and sufficient δ_1 . If we let $\delta_1 = 0.5cm$ we can see that o_1 would be deformed by an extra 0.2cm resulting in possible damage to the triangular part from excessive deformation.

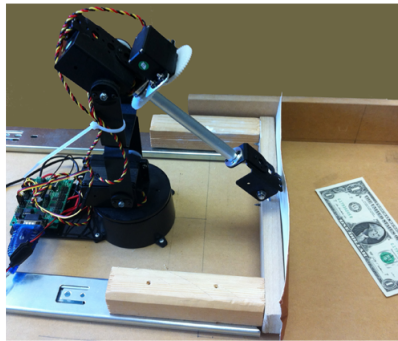
B. Experiments

In this section we describe in detail the experimental setup, example experimental trials, and present the results that will be discussed in the Conclusions.

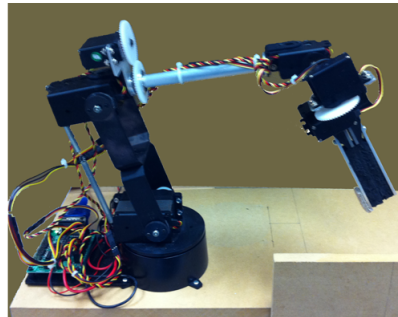
The experimental setup, Fig. 25, utilized two low precision Lynxmotion robot manipulators. The two robot manipulators were set up orthogonal with opposing fixed walls on the other sides of a rectangular workspace that was twenty-nine centimeters square. Following closely the work of [1] we wanted to replicate the effect of using a parallel-jaw gripper; thus, we made a moveable wall (the sweeper) with a parallel fixed-wall and only allowed the sweeper to move orthogonal to its parallel fixed wall. The primary manipulator was set up to utilize only two of its three degrees-of-freedom to move the sweeper and the secondary manipulator was set up with seven degrees-of-freedom and a parallel-jaw gripper that allowed it to perform all of the reorientation actions: PUSH, GRASP, and REORIENT. We implemented inverse



(a) Setup



(b) Primary manipulator



(c) Secondary manipulator

Fig. 25.: The complete experimental setup (a), the primary manipulator with one degree-of-freedom in operation space (b), and the secondary manipulator able to PUSH, GRASP, and REORIENT (c).

kinematics for positioning of the manipulators. To remain sensorless we utilized an open-loop approach to controlling the manipulators. This was achieved by using the inverse kinematics to calculate pulse commands for the manipulator servos that were then sent to the servo controller.

The parts, Fig. 26, used for the experiments were made of card stock, dollar bills, a textile patch, or foam material, each of which was readily available, and proved easy to alter into different shaped parts. Each of the four materials had similar deformation characteristics that allowed us to use the linear tent model, see Adding Deformation, with modified dimension parameters.

Fig. 26 contains nine parts used during the experiments. Of these nine, the algorithm generated *Plans* for six of them, the other three being determined to be infeasible with the devices we employed. These *Plans*, see an example in Fig. 27, were then executed thirty-two times for each part with the part being placed in a different initial orientation and pose each time. The successes, failures, and reasons for failures were recorded and are shown in Table I.

Fig. 28, Fig. 29, and Fig. 30 illustrate the data from Table I as a whole and broken down by part.



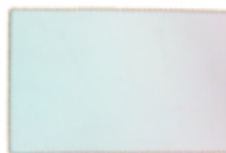
(a) Dollar



(b) Card



(c) Patch



(d) Foam1



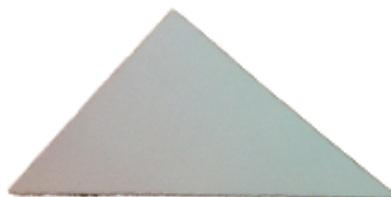
(e) Foam2



(f) Foam3



(g) Foam4



(h) Foam5



(i) Foam6

Fig. 26.: The parts used in the experimental trials.

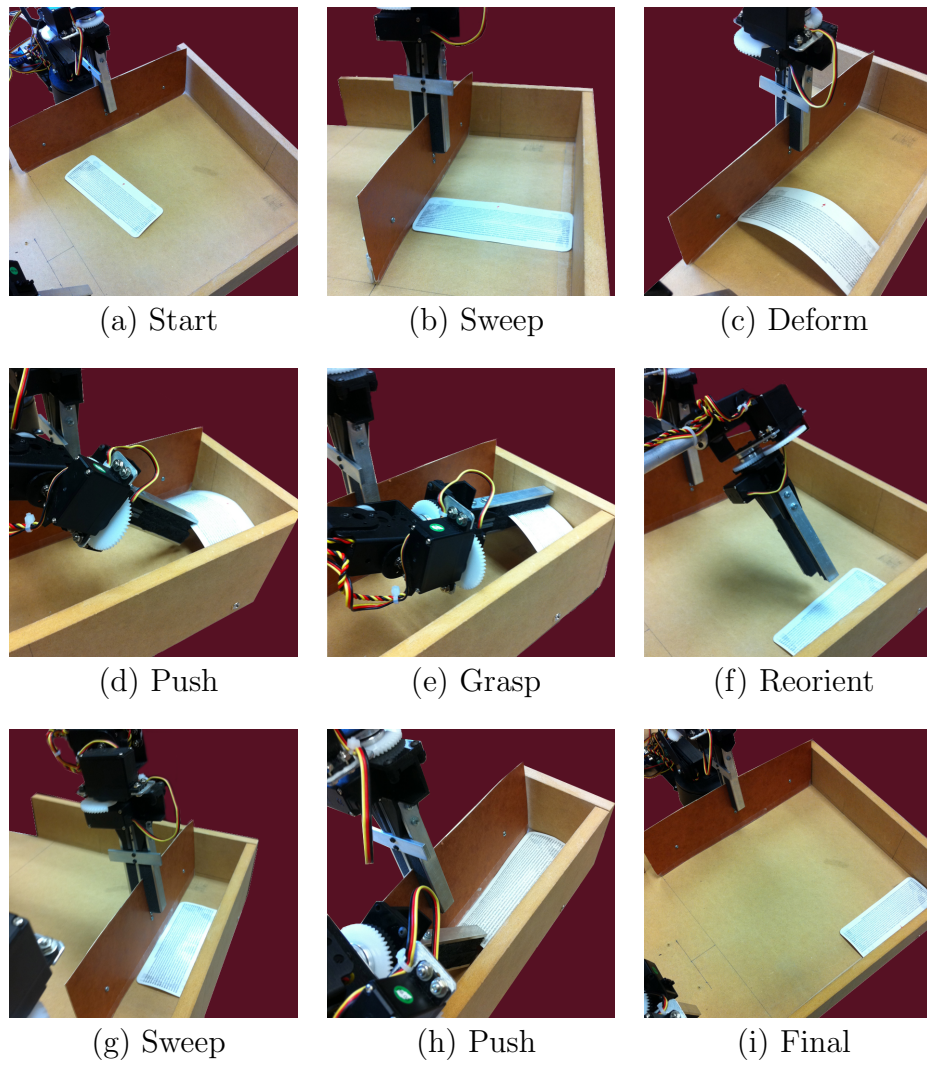


Fig. 27.: Complete plan for Card part.

Table I.: The experimental results of 192 trials of 6 parts.

			Failures			
Part	Material	Complete Successes	Friction	Manipulator Precision	Deformation Model Accuracy	Complete
Card	card stock	22	6	4	0	4
Dollar	dollar bill	7	11	3	11	14
Patch	textile	23	6	1	2	3
Foam1	foam	18	7	5	1	9
Foam2	foam	22	7	3	1	7
Foam3	foam	22	9	1	0	1
Foam4	“Precision insufficient.” Due to overlapping deformation.					
Foam5	“Precision insufficient.” Failed to meet PUSH <i>end-effector constraints</i> .					
Foam6	“Precision insufficient.” Failed to meet REORIENT <i>end-effector constraints</i> .					

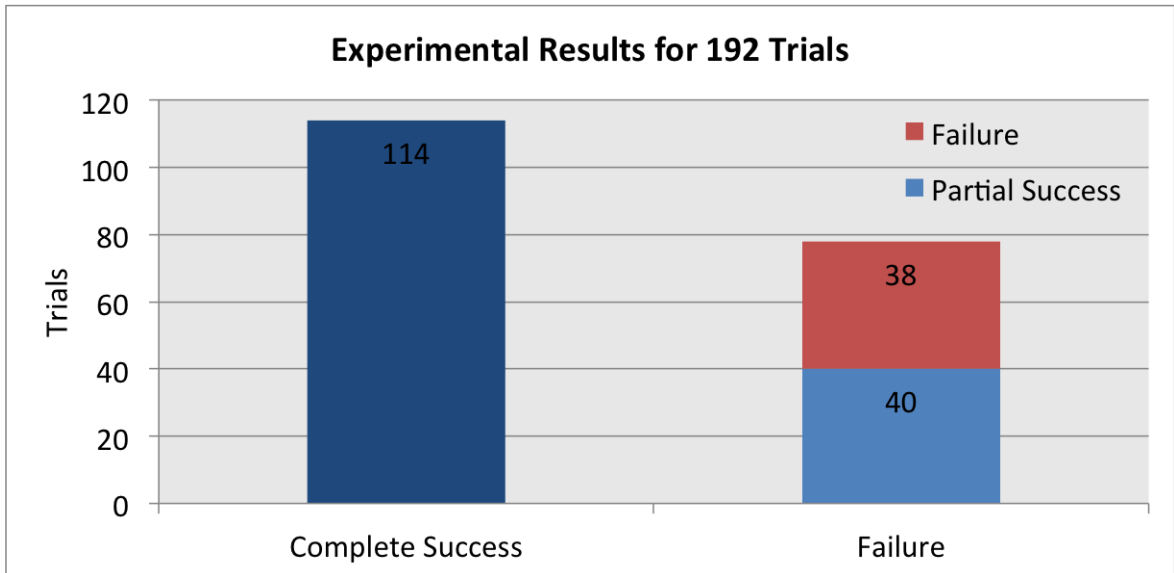


Fig. 28.: The complete successes, partial successes, and failures of the experiments.

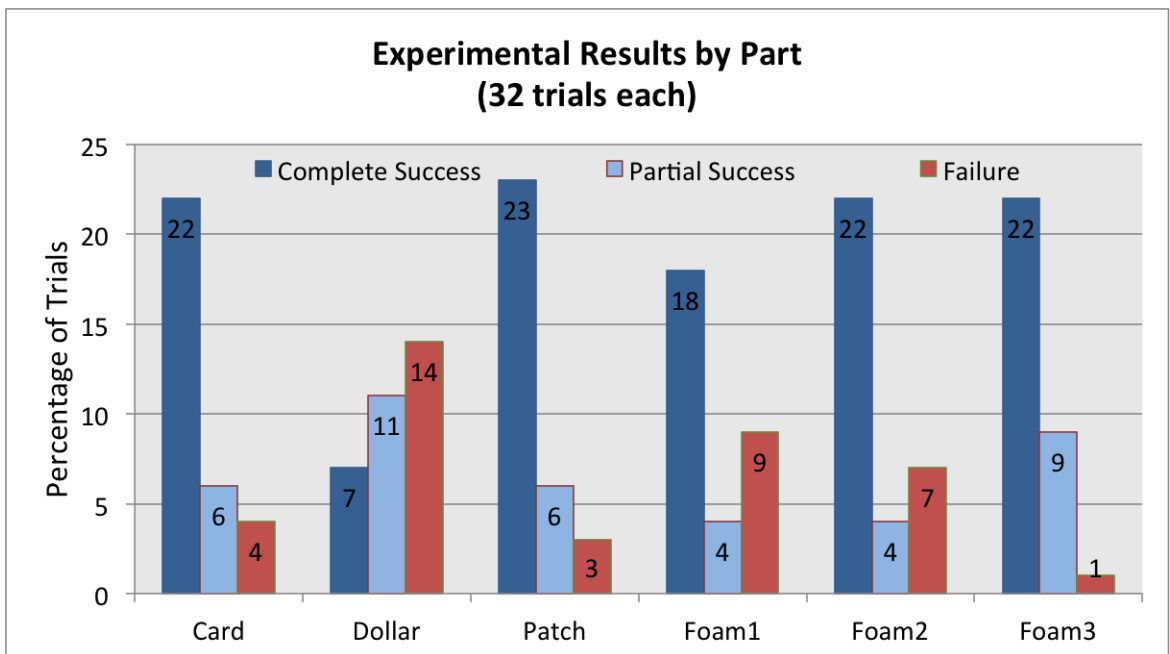


Fig. 29.: The complete successes, partial successes, and failures of the experiments broken down by part.

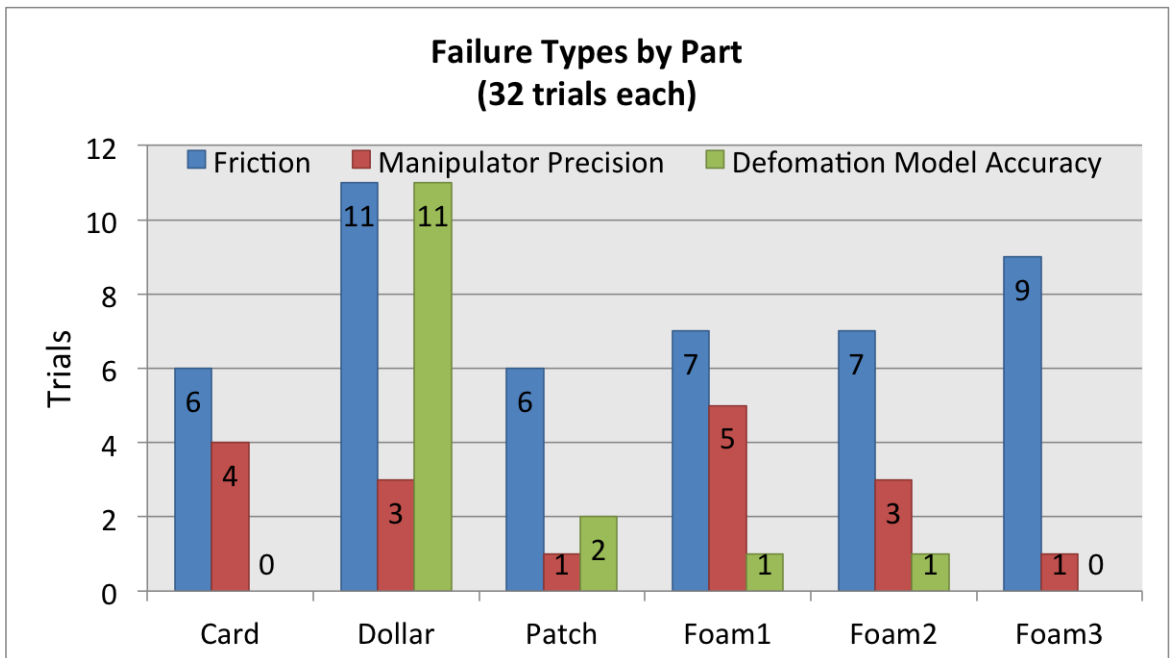


Fig. 30.: The types of experimental failures broken down by part. Includes partial successes and complete failures.

CHAPTER V

CONCLUSIONS

An algorithm has been developed that generates the shortest *Plan* necessary for successful orienting of deformable parts from any initial orientation and pose, and this has been implemented on low-cost (low-precision) manipulators with 60% success. These 114 successes of the experiments show that with the usage of low-precision manipulators and a *deformation model* with low accuracy that orienting deformable parts is possible, and with higher-precision manipulators and a more accurate *deformation model* one can expect increases in the ratio of complete successes to failures.

We presented data from 192 trials of orienting plans for six parts made of four types of materials: paper, textile, card, and foam. Two types of success were defined for improved analysis of the data: complete and partial. Complete success occurs when all assumptions hold true, all actions are successful, and upon termination of the plan the part is oriented and posed correctly. Partial success is a subset of failures that still terminated with the part oriented and posed correctly. Any trial that terminated with the part incorrectly oriented or posed was considered a complete failure.

Defining success and failure in this manner shows the following. First, even with a pessimistic view of the results the approach to orienting deformable parts is possible as shown by the majority of complete success, see Fig. 28. Second, one can see that there is a possibility that the part can be successfully oriented even with failed actions and violated assumptions. This lead to the conclusion that the sweeper and its fixed-wall can be considered a simple mechanical parts-feeder that orients parts by diameter. With further analysis perhaps this could be addressed in future work as an extension for increased success orienting deformable parts.

The failures of each trial were recorded and are analyzed in the following. The first salient point to address in Fig. 30 is that of the failures due to friction. The essential assumption of no friction between the part and parallel-jaws from previous works was found even more important in this work where the part slides along the walls. It is understood that assuming zero friction is optimism that the frictional forces are negligible, but the results show that this assumption can be acceptable under the right conditions. The second leading type of failure is a result of the manipulators having not only low precision, but also non-linear variance of the precision throughout the manipulator’s configuration space. The last type of failure was a result of over simplification of the deformation model for parts such as the dollar that had more variation in the deformation.

Recall from the Introduction that this work took a minimalist approach utilizing simple models, no sensors, and low precision manipulators. Most literature in manipulating deformable objects would lead one to believe that the experimental failures should be dominated by those related to the simplified deformation model or the low precision manipulators, but individually these types of failures were one-third of the failures due to friction. This abundant failure of the expectation that there is no friction between the part and the sweeper reiterates the point made by Mason [5], that expectation is an important means of dealing with uncertainty.

A. Applications to Truly Sensorless Orienting of Rigid Parts

The algorithm has successfully generated plans that guarantee a deformable part will be oriented up to symmetry in its convex-hull. Here we describe possible modifications that would allow the algorithm to orient rigid parts truly sensorless. It was shown that Goldberg’s *stable orientations* are the key to exploiting deformation for

orienting purposes, but in rigid parts there is no deformation. In order to exploit the *stable orientations* for rigid parts we propose the following alterations to this work. First, we replace DEFORM with LOWERWORKSURFACE and reorder the sequence of actions to: SWEEP, PUSH, LOWERWORKSURFACE, GRASP, and REORIENT. LOWERWORKSURFACE along with the assumption vertical friction forces exist between the sweeper and the part, but horizontal frictional forces do not (approximated by use of horizontal roller-bearings) allows SWEEP($Diameter(o_i)$) to apply enough force to hold the rigid part in a *stable orientation* in place for a GRASP and then a REORIENT to occur.

B. Future Work

Recall the $Plan = \{S_{k-1}, D_{k-1}, P_{k-1}, G_{k-1}, R_{k-1}, \dots, S_0, P_0\}$. This plan addresses each of *deformable orientations* once starting with the *deformable orientation* with the largest diameter. Now, recall that the action REORIENT and the need for a manipulator configuration that meets the *end-effector constraints*. If the manipulator and part combination require a reorientation greater than the precision of the manipulator allows, then the algorithm would be forced to return an indication of insufficient precision. Mechanically we propose the use of different methods of grasping, such as vacuum grippers, to allow larger ranges of rotation for reorienting purposes. To provide applicability to more manipulators we propose for future work modifications to the algorithm that would allow reorientation of a grasped part to a $o_i \pm \phi_i$ rather than the fixed $o_0 \pm \phi$. This would likely leave the plan unaffected, but it also has the possibility of lengthening the $Plan$ with repeating sequences of actions such as: $\{S_{k-1}, D_{k-1}, P_{k-1}, G_{k-1}, R_{k-1}, S_{k-2}, D_{k-2}, P_{k-2}, G_{k-2}, R_{k-2}, S_{k-1}, D_{k-1}, P_{k-1}, G_{k-1}, R_{k-1}, \dots, S_0, P_0\}$

Another limitation that occurred during the experiments was that of the manipulator for use in the PUSH. We found that the manipulators were unable to meet the PUSH *end-effector constraints* for larger non-symmetrical parts. To overcome this issue for the manipulators and others we propose the addition of various sized tools or a more adjustable gripper for a larger push contact area. Some parts with non-uniform deformation would also allow separate addressing of symmetrical orientations within o_i by multiple PUSH actions. In order for this to be possible the push contact areas of the different orientations must not have any overlap with the symmetrical orientations. With these alterations to the algorithm it would generate a *Plan* such as: $\{S_{k-1}, D_{k-1}, P_{k-1_1}, P_{k-1_2}, G_{k-1}, R_{k-1}, \dots, S_0, P_0\}$.

REFERENCES

- [1] K. Y. Goldberg, “Orienting Polygonal Parts without Sensors,” *Algorithmica*, vol. 10, no. 3, pp. 201–225, 1993.
- [2] R. C. Brost, “Automatic Grasp Planning in the Presence of Uncertainty,” *International Journal of Robotics Research*, vol. 7, no. 1, pp. 3–17, 1988.
- [3] M. T. Mason, “Manipulator Grasping and Pushing Operations,” Ph.D. dissertation, Dept. of Computer Science, MIT, Cambridge, MA, June 1985.
- [4] M. Erdmann and M. T. Mason, “An Exploration of Sensorless Manipulation,” *IEEE Journal of Robotics and Automation*, vol. 4, no. 1, pp. 369–379, August 1991.
- [5] M. T. Mason, “Kicking the Sensing Habit,” *AI Magazine*, vol. 14, no. 1, pp. 58–59, 1993.
- [6] S. Akella and M. T. Mason, “Posing Polygonal Objects in the Plane by Pushing,” in *IEEE International Conference on Robotics and Automation*, 1992, pp. 2255–2262.
- [7] M. Mani and R. D. Wilson, “A Programmable Orienting System for Flat Parts,” in *North American Mfg. Research Inst. Conf XIII*, 1985.
- [8] M. A. Peshkin, “Planning Robotic Manipulation Strategies for Sliding Objects,” Ph.D. dissertation, Dept. of Physics, CMU, Pittsburgh, PA, November 1986.
- [9] M. A. Peshkin and A. C. Sanderson, “The Motion of a Pushed, Sliding Workpiece,” *IEEE Journal of Robotics and Automation*, vol. 4, no. 6, pp. 569–598, December 1988.

- [10] D. Henrich and H. Wörn, *Robot Manipulation of Deformable Objects*, Springer, London, UK, 2000.
- [11] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, “Cloth Grasp Point Detection Based on Multiple-view Geometric Cues with Application to Robotic Towel Folding,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, pp. 2308–2315.
- [12] S. Hirai, T. Tsuboi, and T. Wada, “Robust Grasping Manipulation of Deformable Objects,” in *Assembly and Task Planning, 2001, Proceedings of the IEEE International Symposium on*, 2001, pp. 411–416.
- [13] Y. J. Lin and T. S. Lee, “Comprehensive Modeling for Simultaneous Position and Force Control of Deformable Manipulators,” in *Decision and Control, 1992., Proceedings of the 31st IEEE Conference on*, 1992, vol. 2, pp. 1887–1892.
- [14] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer, “Elastically Deformable Models,” *SIGGRAPH Computer Graphics*, vol. 21, pp. 205–214, August 1987.
- [15] S. F. Gibson and B. Mirtich, “A Survey of Deformable Modeling in Computer Graphics,” Tech. Rep., Mitsubishi Electric Information Technology Center America, Cambridge, MA, USA, November 1997.
- [16] K. J. Bathe, *Finite Element Procedures*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1996.
- [17] S. Natsupakpong and M. C. Cavusoglu, “Determination of Elasticity Parameters in Lumped Element (Mass-spring) Models of Deformable Objects,” *Graphical Models*, vol. 72, no. 6, pp. 61–73, 2010.

- [18] R. Bartels, J. Beatty, and B. Barsky, *An Introduction to Splines for use in Computer Graphics and Geometric Modeling*, Morgan Kaufmann, Los Altos, CA, USA, 1987.
- [19] G. Farin, *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, Academic Press, Boston, MA, USA, 1998.
- [20] M. T. Mason and K. Y. Goldberg and R. H. Taylor, “Planning Sequences of Squeeze-Grasps to Orient and Grasp Polygonal Objects,” in *Seventh CISM-IFTOMM Symposium on Theory and Practice of Robots and Manipulators*, 1988.
- [21] S. Hirai, H. Wakamatsu, and K. Iwata, “Modeling of Deformable Thin Parts for Their Manipulation,” in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, May 1994, vol. 4, pp. 2955–2960.
- [22] G. Smith, E. Lee, K. Goldberg, K. Bohringer, and J. Craig, “Computing Parallel-jaw Grips,” in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, May 1999, vol. 3, pp. 1897–1903.
- [23] K. Gopalakrishnan and K. Goldberg, “D-Space and Deform Closure Grasps of Deformable Parts,” *International Journal of Robotics Research*, vol. 24, no. 11, November 2005.

VITA

Name: Shawn Michael Kristek

Address: Department of Computer Science and Engineering
Dwight Look College of Engineering
Texas A&M University
3112 TAMU
College Station, TX 77843-3112

Email Address: shawnkristek@tamu.edu

Education: B.S. Electrical Engineering, The University of Texas at Tyler,
2009